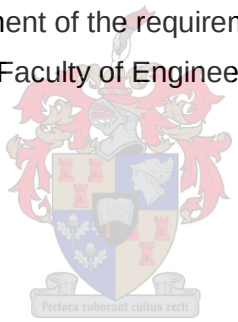
Detection and statistical modelling of output power ramp events for utility scale wind energy facilities

Danielle Lyners

Thesis presented in partial fulfilment of the requirements for the degree of Master of Engineering (Electrical) in the Faculty of Engineering at Stellenbosch University



Supervisor: Prof HJ Vermeulen

March 2021

Plagiarism declaration

Plagiarism is the use of ideas, material and other intellectual property of another's work and to present it as my own.

I agree that plagiarism is a punishable offence because it constitutes theft.

I also understand that direct translations are plagiarism.

Accordingly, all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism.

I declare that the work contained in this assignment is my original work and that I have not previously (in its entirety or in part) submitted it for grading in another project / thesis / dissertation.

Name: Danielle Lyners

Date: March 2021

Abstract

Large and rapid variations in wind power, i.e. ramp events, has received increased attention in recent years. It is generally accepted that accurate forecasting and quantification of ramp events are considered crucial in managing the risks associated with large-scale integration of wind energy resources. The modelling and characterisation of wind power ramp events are of major importance in this context.

The research objectives associated with this project focusses strongly on the investigation of existing ramp models and ramp detection algorithms, as well as the development of improved ramp models and ramp detection algorithms. Some of the most commonly used ramp detection algorithms to date are investigated, including the swinging door algorithm, optimised swinging door algorithm and L1-ramp detect with sliding window. Three new ramp detection models were proposed and investigated, including a multi-parameter segmentation algorithm, a multi-parameter segmentation algorithm with particle swarm optimisation and regression-based segmentation algorithms.

Furthermore, the wind power ramps detected by the multi-parameter segmentation algorithm for optimal parameter values were used to perform statistical analysis of the key ramp features in order to gain insights into wind power ramp events, including the distribution and severity of the ramp events, the frequency of occurrence and seasonality of the ramp events and the distribution of the interarrival times of ramps. The application of cluster analysis to the ramp detection results obtained via the multi-parameter segmentation algorithm for optimal parameter values was investigated to characterise a wind energy facility site in terms of ramping mode. A diverse range of clustering algorithms, distance measures and linkage criteria were investigated to determine the optimal clustering procedure for the data set of interest.

The most important contribution of the work is the development of the multi-parameter segmentation algorithm. Focus was, therefore, placed on evaluating the performance of the multi-parameter segmentation algorithm by comparing its detection behaviour to that of the swinging door algorithm, optimised swinging door algorithm and the L1-ramp detect with sliding window for optimal parameter values. It was concluded that the multi-parameter segmentation algorithm performs significantly better compared to the original swinging door algorithm, as well as similarly or better compared to the L1-ramp detect with sliding window and optimised swinging door algorithm, while also being more computationally inexpensive. The ramp detection performance of the multi-parameter segmentation algorithm is especially superior to the ramp detection performance of the swinging door algorithm, optimised swinging door algorithm and L1-ramp detect with sliding window based on the detection accuracy of the start- and end-points of the ramps, as it correctly identifies the start- and end-points of all the detected ramp events.

Opsomming

Onlangs was meer aandag geskenk aan groot en vinnige veranderinge in windkrag, naamlik helling-gebeurtenisse. Die akkurate voorspelling en kwantifisering van helling-gebeurtenisse word as noodsaaklik beskou ten einde die risiko's verbonde aan grootskaalse integrasie van wind energie bronne te bestuur. Die modellering en karakterisering van helling-gebeurtenisse is dus van groot belang in hierdie konteks.

Die doelwitte van die projek fokus grootliks om die bestaande hellingsmodelle en algoritmes wat helling-gebeurenenisse identifiseer te ondersoek, asook om nuwe hellingsmodelle en algoritmes te ontwikkel wat beter is as die bestaande modelle en algoritmes. Die mees algemeenste algoritmes wat helling-gebeurtenisse identifiseer, naamlik die swaai deur algoritme, die geoptimaliseerde swaai deur algoritme en die “L1-ramp detect with sliding window” was ondersoek. Drie nuwe hellingsmodelle was voorgestel en ondersoek, naamlik die multi-parameter segmenterings algoritme, die multi-parameter segmenterings algoritme met optimering van deeltjieswerm en segmenterings algoritmes wat op regressie gebaseer is.

Die windkrag helling-gebeurtnenisse wat deur die multi-parameter segmenterings algoritme geidentifiseer is vir optimale waardes van die parameters, word gebruik om ‘n statistiese analise van die hoof eienskappe van die helling-gebeurtenisse uit te voer. Die doel daarvan is om insigte oor die windkrag helling-gebeurtense te kry, insluitend die verspreiding en erns van die helling-gebeurtenisse, die seisoengerigtheid van die helling-gebeurtenisse en hoe gereeld dit voorkom, asook die verspreiding van die tydsduur tussen die helling-gebeurtenisse. Die toepassing van groeperings-analise word ondersoek op die helling-gebeurtenisse wat deur die multi-parameter segmenterings algoritme geidentifiseer is, ten einde ‘n wind energy aanleg te karakteriseer in terme van ‘n hellingsmodus. Verskeie groeperings algoritmes, afstand maatstawe en koppelings maatstawe was ondersoek om te bepaal wat die optimale groepering prosedure is vir die spesifieke datastel is.

Die belangrikste bydrae van die projek is die ontwikkeling van die multi-parameter segmenterings algoritme. Daar word dus grootliks gefokus op die evaluering van die prestasie van die multi-parameter segmenterings algoritme deur dit met die swaai deur algoritme, die geoptimaliseerde swaai deur algoritme en die “L1-ramp detect with sliding window” te vergelyk. Die gevolgtrekking is dat die multi-parameter segmenterings algoritme aansienlik beter presteer as die oorspronklike swaai deur algoritme, asook soortgelyk of beter presteer in vergelyking met die “L1-ramp detect with sliding window” en geoptimaliseerde swaai deur algoritme, terwyl die loop-tyd ook aansienlik vinniger is.

Acknowledgements

Firstly, I would like to thank God for His strength, love, encouragement and wisdom to complete the project and report. It is by His grace that I was able to complete my master's degree.

I would like to thank my supervisor Professor HJ Vermeulen who provided insight and expertise that greatly assisted in the development of the work and the report. I really appreciate his valuable guidance, assistance, motivation and patience.

I would like to express my gratitude to my Matthew Groch who provided valuable guidance and input that greatly assisted in the development of the project and the report, and for his patience and willingness to help.

I would like to express my gratitude to CSIR and the Department of Science and Technology (DST) who awarded me a bursary to pursue a master's degree in wind energy.

Finally, I would also like to thank my family and friends for their support, especially my sister Monique Lyners and my friend Obakeng Letlhage for proofreading my report.

Table of Contents

List of figures

List of tables

List of abbreviations

ANOVA	Analysis-of-Variance
CSI	Critical Success Index
CSP	Concentrated Solar Power
DIANA	Divisive Analysis
EAF	Energy Availability Factor
EECP	Emergency Electric Curtailment Plan
ERCOT	Electric Reliability Council Of Texas
FBS	Frequency Bias Score
FN	False Negative
FP	False Positive
GGMM	Generalised Gaussian Mixture Model
IRP	Integrated Resource Plan
L1-SW	L1-Ramp Detect with Sliding Window
OLS	Ordinary Least Squares
OpSDA	Optimised Swinging Door Algorithm
PAM	Partitioning Around Medoids
PDF	Probability Density Function
POD	Probability of Detection
PSO	Particle Swam Optimisation
PV	Photovoltaic
SDA	Swinging Door Algorithm
SO	System Operator
SR	Success Ratio
sYeY	start-Yes-end-Yes
sYeN	start-Yes-end-No
sNeY	start-No-end-Yes
sNeN	start-No-end-No
REIPPPP	Renewable Energy Independent Power Producer Procurement Plan
RMSE	Root Mean Square Error
TP	True Positive

TN	True Negative
VRE	Variable Renewable Energy
WASA	Wind Atlas of South Africa
WPRE	Wind Power Ramp Event
WSS	Within cluster Sum of Squares

1 Project Motivation and Project Description

1.1 Introduction

The development of renewable energy has become the main consensus and strategy globally to address the energy crisis and climate change concerns [1], [2]. Wind power, in particular, is one of the most important renewable energy sources due to the abundance of the resource as well as the sustained technological progress resulting in economic affordability and technological maturity [3].

South Africa already has a sizeable installed capacity of wind energy and this value is expected to increase in the future. The South African energy mix consists of 38 GW installed capacity from coal, 3.8 GW from diesel, 3.7 GW from renewable energy, 2.7 GW from pumped storage, 1.8 GW from nuclear and 1.7 GW from hydro [4]. The addition of new and clean generating capacity is urgently required because of factors such as the current generation capacity constraints, the expected reduction in the availability of existing generating assets due to an ageing coal-fleet, the reduction of the Energy Availability Factor (EAF) of existing assets, the growing electricity demands and the international climate change commitments [4], [5], [6]. The potential for wind and solar power in South Africa is significant as South Africa has excellent wind and solar resources, as well as the benefit that the solar and wind power profiles match relatively well with the demand profile [5]. The global advances in wind and solar power has, furthermore, made it possible to incorporate these renewable energies without greatly compromising economic objectives [6]. Wind and solar power can also be used to provide off-grid electricity [4].

A number of policies and policy instruments was introduced to promote the development of renewable energy projects in South Africa, including the White Paper on Renewable Energy in 2003 [7], the Integrated Resource Plan (IRP) in 2011 [8], the IRP in 2019 [4] and the Renewable Energy Independent Power Producer Procurement Program (REIPPPP) in 2011 [9]. The IRP is a development plan for the electricity infrastructure of South Africa. The objective of the IRP is to balance electricity generation and supply at minimum cost, while considering energy security and the environment [4]. The most recent IRP was published in 2019 and describes the optimal generation mix that is required to service the forecasted demand growth until 2030. The IRP is to be reviewed and updated regularly, at least every two years, to take changing circumstances such as technological progress and the impact of these advances into account when choosing the optimal technology mix [8]. The REIPPP is a competitive tender process that promotes and acquires investments in grid-connected renewable energy projects from private entities. Preferred bidders must sign power purchase agreements with Eskom, with the tariffs being subjected to inflation [10]. The renewable energy technologies that can be selected under the REIPPP include wind, solar Photovoltaic (PV), Concentrated Solar Power (CSP), small hydro, biomass and landfill gas. The REIPPP has been successful with regards to reducing the cost of renewables as well as reaching capacity outcomes [10]. A total of 6422 MW of generation capacity has been secured to date via the REIPPP [4] across 7 bid windows and 112 projects [11], of which 3876 MW is already operational and connected to the grid [4].

A large increase in the installed wind power capacity is expected in the future. This will increase the percentage of electricity generation from wind power, as well as the impacts of wind power on grid operations [12]. Despite the advantages that wind power has to offer, the variability and uncertainty of the wind power creates challenges to the system operator for the large-scale integration of wind power, especially with reference to maintaining power balance and in terms of the ramping requirements imposed on conventional generation units. This variability and uncertainty may result in increased

ancillary service requirements to maintain balance between supply and demand [13]. The extent to which the variability and uncertainty of wind power will impact the performance of the overall power system is influenced by the generation capacity, capacity factor and capacity credit, as well as the demand profiles [6].

It is important to obtain a better understanding of wind power variability to be able to manage the impacts it has on the power system and consequently enable successful integration [14]. Accordingly, it is necessary to conduct the relevant power system studies, which creates a need for representative wind power data [15]. The Wind Atlas of South Africa (WASA) makes it possible to quantify the potential for wind power generation at a specific site in South Africa. To date, wind projects has mostly been developed in the Western Cape and Eastern Cape [4].

1.2 Project motivation

The power generation profiles associated with wind power sources exhibit a relatively high degree of uncertainty and variability [16]. This presents major technical and economic challenges to power system operators (SOs), especially as the installed capacity increases with increased penetration [17], [18]. These challenges include maintaining grid stability, controlling the system frequency and maximizing the economic and environmental benefits of the renewable energy resource [18].

Power systems generally have the ability to manage small amounts of variability and uncertainty [19]. In practice, small amounts of variability in the residual load profile are generally handled by generator control mechanisms [17]. Extreme ramp events are, however, critical [17] as it may require the dispatch of rapid response generation, or the implementation of load shedding to protect the stability of the system [17]. Ramp events can therefore have significant impacts on the reliability and economics of the system, which is of primary importance to system operators [19].

From a research perspective, it is therefore important to focus on extreme events, i.e. events that have the potential to significantly impact the reliability, security and economics of the power system [16], [20]. Wind power ramp events are defined as significant positive or negative changes in wind power output in a limited period of time [12], [21], and are associated with weather phenomena such as intense low-pressure systems, thunderstorms, low level jets, wind gusts or other similar physical processes [20], [21]. Wind power ramps are affected by different time and geographic scales [19].

It is generally accepted that accurate forecasting and quantification of ramp events are considered crucial in managing the risks associated with large-scale integration of wind energy resources [12], [18]. Quantifying the severity and frequency of wind power ramps can aid in the planning of power system operations, as this information can be used to anticipate the necessary control actions [22].

1.3 Project description

1.3.1 Research objectives

The research objectives of the project can be summarised as follows:

- Review the existing models to describe and detect ramp events.
- Investigate the development of improved ramp models and ramp detection algorithms that are closely aligned with the grid impacts induced by variable wind power. The modelling approach should make allowance for short- and long-term impacts of ramping under conditions of variable penetration of wind power.

- Evaluate and compare the performance of the existing and proposed models and ramp detection algorithms, based on the metrics commonly applied in literature.
- Perform statistical analysis of the key ramping features with the view to obtain insights into wind power ramp events that can be used to make informed scheduling decisions as well as to develop forecasting models for ramping.
- Develop a methodology to characterise existing and potential wind energy facility sites in terms of ramping mode. The proposed methodology should make use of historical data from meteorological measurements masts where available. It is envisaged that the model may be of use in the medium- and long-term planning of wind energy deployment.

1.3.2 Key questions

The research objectives give rise to the following key questions to be investigated and answered during the project:

- What ramp models and ramp detection algorithms have been proposed to date?
- Can an improved ramp model and ramp detection algorithm be developed?
- What is the best approach to characterise the statistical properties of ramp events?
- What impacts do wind power penetration, specifically with reference to wind power ramps, have on the grid? Why are wind power ramps of primary concern to system operators?
- Can the application of cluster analysis to the ramp detection results be used to characterise a wind energy facility site in terms of ramping mode?
- What is the optimal clustering procedure to group wind power ramp events?

1.4 Thesis structure

The remainder of this report is organised as follows:

- *Chapter 2: Literature Review*

  A literature study on the relevant topics is presented.

- *Chapter 3: Ramp detection models*

  Detailed descriptions of the existing and proposed ramp detection models are provided, and the corresponding results are presented. A comparison of the main ramp detection models is provided.

- *Chapter 4: Statistical analysis*

  A statistical analysis of the key ramp features as obtained for an optimal ramp detection model is presented.

- *Chapter 5: Clustering analysis*

  A detailed description of the clustering methodology applied to wind power ramps as obtained for an optimal ramp detection model is provided.

- *Chapter 6: Conclusions and recommendations*

  Final conclusions regarding the research are presented and recommendations for future work are presented.

2 Literature Review

2.1 Overview

The ramping phenomena associated with wind power generation has been studied extensively in recent years. This project therefore required an in-depth review of the literature pertaining to the subject, especially with reference to the following:

- The temporal characteristics of wind generation power profiles.
- The impacts of variable and uncertain wind generation power profiles on grid operations.
- Ramping models.
- Ramp detection algorithms.
- Ramp classification methodologies and metrics.

The research objectives associated with the project focuses strongly on the development of improved ramp models and ramp detection algorithms, as well as the statistical analysis of ramp events. This work required the use of various machine learning methodologies, including the following:

- Regression analysis.
- Particle swarm optimisation.
- Clustering algorithms.

2.2 Overview of wind power characteristics

2.2.1 Implications of wind power generation for systems operations

The reliable operation of the power system requires that the frequency and the voltage of the system is maintained within acceptable levels and that supply and demand is balanced at every time instant [23]. This needs to be ensured during all operating conditions, including unexpected failure of equipment, fluctuations in demand or supply, maintenance schedules, and respecting the technical limitations of the conventional generation units [5].

Fig. 1 depicts power system dispatch operations with and without Variable Renewable Energy (VRE) [24]. Wind power differs from conventional generation units in the sense that it is not dispatchable. It is therefore often modelled as a negative load in the power system [25]. The integration of wind power into the power system thus means that the residual load, i.e. the load minus the output of renewable sources, must be balanced by the conventional generation units. The conventional generation units are required to adjust their output to the residual load, either by starting or stopping the generation units, or by changing the power output while synchronised [24]. Generation units, however, have technical limitations such as start-up and shut-down times, minimum uptimes, minimum stable operating levels and ramp-rate limits [24]. The operating cost of the different generation units varies [24]. It is thus necessary to plan the dispatch of the generation units in advance in order to optimise the cost and to ensure that system balance and security are maintained [24], [25]. System planning requires accurate forecasts of demand and generation. In this context, reliable and accurate wind power forecasts are crucial for successful integration of wind power into the grid [18]. Since forecasts are not 100% accurate, flexible rapid response balancing reserves are required to balance the mismatch between the forecasted residual load and the actual residual load as well as the generation [24], [26].

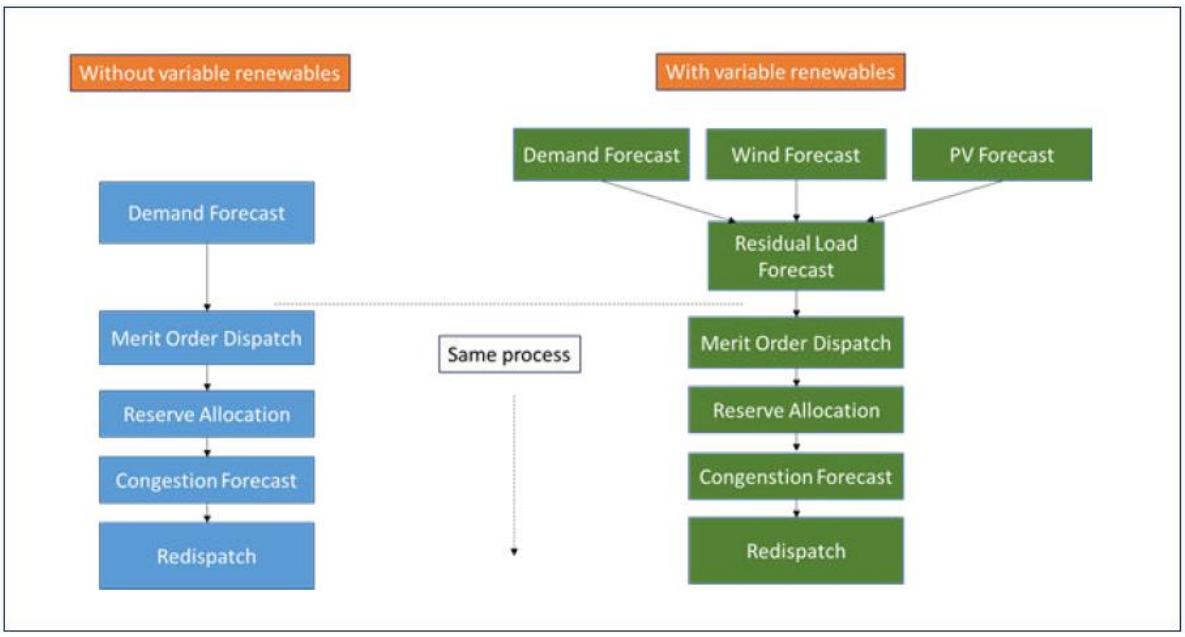


Fig. 1: Power system dispatch operations with (Right) and without variable renewable energy (Left) [24].

2.2.2 Variability and uncertainty

The power generation profiles associated with wind power sources exhibit a relatively high degree of variability and uncertainty [16], [18], [23].

- *Variability*: Wind power is not constant, but varies with time according to the availability of wind [23], [27]. Wind power production is variable over all time scales, i.e. seconds, minutes, hours, days, months and years [18].
- *Uncertainty:* Wind is not controllable, and it cannot be scheduled. It can only be forecasted, as in the case of the load [24], [28]. Wind power, furthermore, is uncertain because it does not fluctuate in a discernible pattern, which results in limited predictability [27], [29]. Wind power is uncertain over multiple time scales [21].

The variability and uncertainty inherent to wind power presents major technical and economic challenges to power System Operators (SOs), especially as the installed capacity increases with increased penetration [17], [18], [20], [23], [27], [29], [30]. These challenges include maintaining grid stability, controlling the system frequency and maximising the economic and environmental benefits of the renewable energy resource [18].

It is important to note that variability and uncertainty is not a new phenomenon in power system management, as demand is also not constant, but fluctuates over time [27]. The integration of wind power, however, leads to an increased variability and uncertainty in the residual load because variability and uncertainty in demand is now integrated with variability and uncertainty in generation [16], [20], [25]. The variability in the residual load will be exaggerated in terms of frequency of occurrence and rate of change [27]. Specifically, more extreme variability is present in the residual load compared to the load because of the extent of variability present in the wind power output, and because variations in the load and wind power can occur simultaneously in opposite directions [27], [31]. The residual load is consequently less predictable than the load because of the integration of wind power. Wind power is, furthermore, less predictable compared to the load, because wind power does not fluctuate in a

discernible pattern while load is more stable. Fast fluctuations are also more difficult to predict [20], [27], [32].

The increased variability and uncertainty in the residual load must be met by the conventional generation units [25]. The increased variability imposes additional ramping and cycling requirements on these units, because it must adjust its output more often to correspond to fluctuations [26], [27], [29]. The conventional generation units must therefore be capable of higher production ramping rates and a greater range of capacity output [25], [29], [33]. Uncertainty complicates the scheduling of generation units [32]. The increased uncertainty necessitates the use of additional short-term balancing services and the provision of operating reserve capacity to ensure reliable power system operation [26], [29], [32]. It follows that the uncertainty associated with wind power production causes significant cost to be associated with the integration of wind power into the grid [21]. It becomes more challenging to respond to variability during increased uncertainty [16].

Nonetheless, the general consensus is that the challenges associated with the integration of wind power are manageable. Additional integration cost is, however, introduced at system level [29]. Wind power forecasting is crucial to the successful integration of wind power into the grid, especially as the installed capacity increases with increased penetration, because it minimises the uncertainty associated with wind power [19].

Among the challenges introduced by the variability and uncertainty of wind power, the ability to handle large wind power ramp events is of great concern to power system operators [19], [34]. This is corroborated by the increasing number of studies on wind power ramp events [13]. Wind power ramp events are defined as significant positive or negative changes in wind power output in a short period of time [12], [21]. Wind power ramp events are considered to be those events that have the potential to significantly impact the reliability and economics of the system, which is of primary importance to the system operators [19], [35].

2.3 Wind power ramp events

2.3.1 Overview

Ramp events can generally be divided into two basic types based on the direction, namely up-ramps and down-ramps. Up-ramps are defined as rapid increases in wind power, which are induced by strong low-pressure air systems (or cyclones), low-level jets, thunderstorms or similar atmospheric phenomena. Down-ramps are defined as rapid decreases in wind power. Down-ramps are the reverse physical processes of up-ramps and are generally caused by the same aforementioned phenomena or it is caused by high speed gusts of wind which make wind turbines reach cut-out limits and shut down to protect themselves from damage [18], [20], [21].

2.3.2 Impacts of ramp events on power system operations

The different strategies employed to manage wind power ramp events are subject to the characteristics of the ramps and the timing of the ramps in relation to other events, as well as how far in advance the ramps are predicted [27], [31]. The characteristics of the ramps determine the capacity and ramp rate requirements imposed on the conventional generation units to compensate for the ramp [36]. Generally, an up-ramp must be compensated for by reducing the output of other generation units. This can be done by ramping these units down or shutting them off, or in certain cases by curtailing the wind power output. Conversely, a down-ramp must generally be compensated for by increasing the output of other

conventional generations units. This can be done either by ramping up online-units or starting up offline-units. Evidently, it is necessary to ensure that enough back-up generation is available. In extreme cases, i.e. when the latter measures are not sufficient, the implementation of load shedding may be required to protect the stability of the system [18], [20], [36], [37]. The management of ramps is done via expensive ancillary services [21]. Accurately predicting wind power ramp events well in advance will allow system operators to make optimal management decisions to ensure reliable power system operation and maximise economic and environmental benefits [18].

Severe grid management issues arise if wind power ramp events occur without prior warning [38]. Power systems generally have the ability to manage small amounts of variability and uncertainty [16], [19], [34]. In practice, small amounts of variability and uncertainty in the residual load profile are generally handled by generator control mechanisms, while large wind power ramp events may require re-dispatching the system or even load-shedding [17], [35]. In addition to wind power ramps representing significant variability, it is also associated with significant uncertainty because of the difficulty in accurately predicting these events [21], [28]. Therefore, large forecast errors may result in the need for expensive short-term balancing services or load shedding to maintain the balance between supply and demand [28].

Ela and Kirby [39] reported the occurrence of a large down-ramp event on 26 February 2008 in the Electric Reliability Council of Texas (ERCOT) that required ERCOT to call in the Emergency Electric Curtailment Plan (EECP). These measures were necessary because it became increasingly challenging to maintain supply-demand balance, resulting in the reduction of the system frequency. The factors which contributed to the worsening imbalance include the occurrence of a wind power ramp event, the unexpected loss of a generation unit, as well as the demand rising more sharply than was predicted. ERCOT was required to employ reserve capacity to increase generation as well as decrease demand. It was concluded that if accurate forecasts of the generation and demand were provided, more informed generation scheduling decisions could have been made, which in turn could have prevented the need for the emergency response [39]. This particular situation reiterates the importance of having information about the timing of the ramp in relation to other events, because the impact of the ramp event greatly depends on the ramp timing relative to other events [31]. Specifically, if wind power ramp events coincide with ramping of the demand in the opposite direction, it will result in more severe ramping in the residual load [20].

In summary, wind power ramp events are critical because of the significant uncertainty associated with it as well as the risk that the wind power ramp event poses to the management of the power system [40]. The mismanagement of wind power ramp events will result in more substantial damage to the power system as well as increased cost, when compared to the mismanagement of a non-ramp event [35]. Furthermore, the impact of wind power ramp events on the grid increases as the magnitude of the ramp increases or the duration of the ramp decreases [12], [31]. Additionally, as the wind power penetration level increases, the intensity and frequency of ramping events also increases. This in turn increases the impact that wind ramp events have on the grid [12], [20].

Wind power ramp events have short- and long-term effects on the grid due to requirements imposed on conventional generation sources. In the medium- to long-term scenario, excessive ramping gives rise to premature aging of thermal plant, leading to shorter maintenance cycles and increased maintenance costs. In the short-term scenario, ramping complicates the optimal dispatch of conventional generation plants and imposes increased reserve margin requirements, which is costly [24].

2.3.3 Managing the grid impacts of wind power ramp events

Various strategies exist that can mitigate the impact of wind power ramp events and facilitate the successful integration of wind power to the grid. Not all options are, however, viable in practice due to economic or political reasons [28]. It is generally accepted that accurate forecasting of ramp events and assessment of the forecast uncertainty as well as quantification of ramp events, i.e. obtaining a better understanding of ramp events, are crucial in managing the risks associated with large-scale integration of wind energy resources into the grid [12], [18].

The forecasting accuracy of wind power ramp events significantly impacts the effectiveness and cost-effectiveness of the provision of warning information in advance so that optimal scheduling decisions can be made [41]. Predicting the magnitude, duration and rate of a ramp can determine the capacity and ramp rate requirements imposed on the conventional generation units to accommodate the ramp and maintain reliable system operation. With regards to cost-effectiveness, it is generally cheaper to run conventional generation units with longer start times. Thus, it is useful to accurately predict the event in advance. Additionally, it is generally cheaper to run conventional generation units with longer minimum on times, considering the unit will run for the entire minimum on-time. It is, therefore, crucial to predict the duration of the ramp as well as how long the energy level is sustained after the ramp has ended [36]. Furthermore, increasing the accuracy and reliability of forecast systems can contribute to the advancement of an automated system created to manage wind power ramp events. Such a system could automatically activate reserves, reduces or increases the output of conventional sources, or curtails the output of a wind farm in response to the ramp event [42].

An effective forecasting system will provide reliable and accurate information about the general wind power behaviour and in particular wind power ramp events [28]. Traditional forecasting methodologies aim to provide accurate point forecasts of wind power. These methodologies are optimised by minimising bulk error metrics such as Root Mean Squared Error (RMSE), which are simple metrics [28]. These bulk error metrics penalises incorrect predictions of ramp timing heavily. Consequently, the methodology predicts smoother ramps in order to cover a wider range [13]. The output of the forecast methodologies therefore tends to the mean output and fails to capture wind power ramp events [16], [28]. In addition, since wind power ramp events occur relatively rarely, it generally represents outliers in model training data. Consequently, when modelling forecast relationships from a standard training sample, little weight is given to wind power ramp events [43]. In view of the above, a forecast system dedicated to wind power ramp events is required. Significant uncertainty is, however, still associated with wind power ramp events when using a dedicated forecast system [28]. Even with significant forecast uncertainty the wind power ramp forecasts can still be valuable to a power system operator [28].

Accurate forecast of wind power ramp events, as well as assessment of the forecast uncertainty associated with these events, contribute to the successful large-scale integration of wind power into the grid [12]. It is generally accepted that probabilistic forecasts are useful to system operators, as it provides information about the uncertainty associated with wind power events. This information will enable system operators to define acceptable levels of risk and make optimal scheduling decisions [28]. Operating procedures entails that some level of risk will always be acceptable in order to limit the cost of ancillary services [28]. An acceptable level of risk is defined by factors such as the probability of the event and the cost associated with the events [28].

Obtaining a deeper understanding of ramp events and the characteristics of its features can provide additional insights that are helpful in making informed scheduling decisions to mitigate the impact of wind power ramp events [21], [37], [44], [45]. Statistical analysis of historical or forecasted wind power data can, furthermore, provide additional information about wind power ramp events [44], [46]. Statistical models of wind power ramp events provide valuable insights that can contribute to the development of forecasts and stochastic control strategies. The California Independent System Operator Balancing authority, for instance, have used information about certain ramping features to assist in system scheduling [33].

2.4 Wind power ramp event modelling and detection

2.4.1 Overview

The successful quantification, forecasting and impact analysis of Wind Power Ramp Events (WPREs) requires a formal characterisation of WPREs. In order to accomplish this, it is necessary to derive an appropriate definition as to what constitutes a significant wind power ramp event, as well as a detection algorithm that can extract these ramping events from a temporal wind power profile. One of the major problems associated with wind power ramp event studies is defining a wind power ramp event [18].

Ramp detection is an essential part of forecasting and the quantification of wind power ramp events. The development of statistical models requires robust detection schemes for identifying wind ramps in data. The effectiveness of the statistical models is dependent on the accuracy of the detection algorithms [21]. Ramp detection is used in forecasting to develop metrics that can specifically evaluate the ramp forecasting performance which in turn can be used to tune the ramp forecasting model [13], [16]. Additionally, the ramping events detected from historical data can be used as training data in statistical and machine learning methods to directly predict future wind power ramping events [20]. Ramp detection also enables the analysis of the weather phenomena causing the wind power ramps, which provides insights into the development of forecasting models [35]. Furthermore, it is important that the ramp definition and the detection algorithm are robust to the effects of insignificant fluctuations [21]. The detection algorithm should also be computationally inexpensive to enable online applications [16].

2.4.2 Wind power ramp event definitions

One important question in wind power ramp events studies is how to define a ramp. A ramp event is generally defined as a significant change in power output over a short duration [12], [18], [21], [35]. Wind power ramp events are, furthermore, considered to be those events that have the potential to significantly impact the reliability and economics of the system [19], [35]. Fig. 2 shows a sample wind power signal with an identified ramp event. Several parameters used to characterise a ramp event are indicated [21].

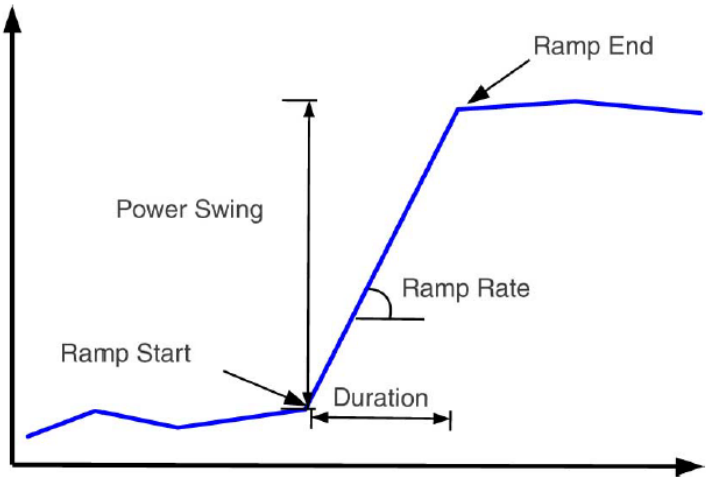


Fig. 2: Sample wind power signal with an identified ramp event. The characteristics of the ramp event, namely the ramp start and end, ramp magnitude, ramp duration and ramp rate, are indicated [21].

According to Fig. 2 as well as other literature, the parameters used to characterise a wind power ramp event are as follows [21], [35]:

- *Magnitude (Power swing):* The magnitude of the wind power change.
- *Duration:* The time span during which the change in wind power occurs.
- *Direction:* Whether the ramp event represents an increase or decrease in power.
- *Timing:* The time of the start or the central point of the ramp.
- *Rate:* The ratio of the ramp magnitude to the ramp duration. This parameter provides an indication of the intensity of the ramp.

These parameters can easily be evaluated if the wind power ramp event is identified [35]. Additionally, it can be used to define wind power ramp events [18].

Although it is relatively easy to identify ramps by visual inspection, there is limited consensus in literature regarding a set definition of a wind power ramp event. Various formal and informal definitions of wind power ramp events have been proposed in literature [12], [14], [18], [21], [38], [46]. Binary classification is often employed to identify wind power ramp events [35]. The binary classification is based on various ramp definitions [12], [18], [28]. It is recognised that different ramp definitions results in the identification of different sets of ramps [21].

The definitions reported to date typically depend on a user defined set of rules aimed at quantifying ramp characteristics, namely the magnitude, duration, direction and rate of the ramp [12], [18], [28]. These characteristics are generally defined with reference to factors such as the size and location of the windfarm, as well as the context of application by the system operator [18]. The magnitude of a ramp can be defined as a fixed value or a percentage of the installed capacity. The fixed value is system-dependent and is generally defined according to technical criteria such as the amount of capacity reserve that is available to backup generation, or as the amount of generation capacity that the system operator struggles to procure to maintain reliable system operation. When using a fixed value, it is likely that more ramps will be detected when the wind power penetration level increases. When defining the magnitude as a percentage of the installed capacity, it is required that the value of the installed capacity is known.

This definition introduces challenges because the installed capacity is not constant. Consequently, the ramp magnitude and in turn also the set of detected ramps varies with the installed capacity. As a result, ramps which have the potential to impact the power system might not be detected [14], [18], [35], [37], [46]. Ramp magnitude can also be defined as a percentage of the standard deviation of the first difference wind power time series or it can be defined so that ramp events occur a certain percentage of the time, e.g. 5% of the time [35]. Additionally, a time-varying threshold can be employed, i.e. the threshold is a percentage of the current power being generated. This approach, however was concluded to be unnecessarily complicated [35]. With regards to ramp duration, the variation in wind power can take place over different time scales. The time scales of interest range from 10 to 60 minutes or one to several hours [17], [37], [47], [40].

Ferreira *et al.* [18] and Gallego-Castillo *et al.* [35] summarised some of the commonly used definitions of wind power ramp events found in literature. Sevlian and Rajagopal [21] reformulated the ramp event definitions initially proposed by Kamath [14], and Zheng and Kusiak [47], into a set of three rules, each of which returns a one if the associated ramp condition is met, and zero otherwise.

The first rule, $R_0(t_i, t_j)$, is defined by the relationship

$$R_0(t_i, t_j) = 1_{\{P(t_j) - P(t_i) > P_{val}\}}, \tag{1}$$

where t_i and t_j denote the i^{th} and j^{th} time instants respectively, and $P(t_i)$ and $P(t_j)$ denote the power at time instants t_i and t_j respectively. The rule determines whether the power has increased by a threshold value P_{val} in the interval defined by t_i and t_j.

The second rule, $R_1(t_i, t_j)$, is defined by the relationship

$$R_1(t_i, t_j) = 1_{\{max(P(t_i) \cdots P(t_j)) - min(P(t_i) \cdots P(t_j)) > \alpha\}} \tag{2}$$

The rule determines whether the difference between the maximum and the minimum power observed in the interval defined by t_i and t_j exceeds the threshold denoted by α. Kamath [14] proposed the ramp definition given by $R_1(t_i, t_j)$ because the ramp definition given by $R_0(t_i, t_j)$ only takes the power at the start and end of the interval into account and so disregards the ramp events present within the interval. Additionally, it was reported that the ramp definitions given by (1) and (2) provided similar ramp statistics [14].

The third rule, $R_2(t_i, t_j)$, is defined by the relationship

$$R_2(t_i, t_j) = 1_{\left\{\frac{P(t_j) - P(t_i)}{t_j - t_i} > \alpha\right\}}. \tag{3}$$

The rule determines whether the rate of increase in the power in the interval defined by t_i and t_j exceeds the threshold denoted by α.

Sevlian and Rajagopal [21] proposed an auxiliary rule to the rules defined by (1) to (3) in order to ensure that large power fluctuations within in the ramp interval cause early termination of the ramp segment. This rule, $R_c(t_i, t_j)$, is defined by the relationship

$$R_c(t_i, t_j) = \prod_{m=i}^{j} \mathbf{1}_{\{P(t_m) > \beta\, max(P(t_i),\ldots,P(t_m)),\ \ \beta < 1\}}, \tag{4}$$

where β denotes the threshold of the fluctuation. This tracks the current maximum value of an interval, which increases iteratively in length, and determines whether the last value of that interval is always greater than a constant value β multiplied by the maximum value recorded for the interval, where $\beta < 1$. $R_c(t_i, t_j)$ returns a one while the rule is satisfied, and a zero otherwise.

The above rules can be combined into a generalised rule, given by the formulation [21]

$$R(t_i, t_j) = R_c(t_i, t_j) \prod_{\tau=1}^{M} R_\tau(t_i, t_j), \tag{5}$$

where $R_c(t_i, t_j)$ is given by (4), $R_\tau(t_i, t_j)$ represents any ramp definition and M denotes the number of rules that must be satisfied to denote a valid ramp.

A range of further ramp definitions has been proposed in literature. Kamath [46] proposed a ramp definition based on the slope of the energy over a fixed time interval. A ramp is identified if the average slope of the intervals between T and $(T + \Delta T)$ exceeds a threshold Tr_{sl}, as formulated in the relationship

$$\left(\frac{1}{n}\sum_{T_i \epsilon [T, T+\Delta T]} sl(T_i)\right) > Tr_{sl}, \tag{6}$$

where n represented the number of intervals present in $[T, (T + \Delta T)]$, and $sl(T_i)$ represents the slope of the i^{th} interval. The slope of the i^{th} interval is calculated via central differencing using the relationship

$$sl(T_i) = \frac{1}{2}\left(MW(T_{i+1}) - MW(T_{i-1})\right). \tag{7}$$

Calculating the slope using (7) may result in increased sensitivity to noise in the data. It may be beneficial to filter the data prior to applying the definition [46]. Furthermore, the threshold Tr_{sl} can be calculated as the linear change of magnitude T_{rmw} over a duration ΔT, as given by

$$Tr_{sl} = \frac{T_{rmw}}{\Delta T}. \tag{8}$$

Bossavy *et al.* [31], [40] proposed a ramp definition that incorporated filtering based on signal processing literature on edge detection. This definition was proposed to reduce the sensitivity of the ramp identification to fast fluctuations or noise which occur in time scales shorter than the ramp durations [35]. The wind power ramps are defined in terms of a filtered version of the wind profile, $P^f(t)$, obtained by the formula

$$P^f(t) = mean\left\{\begin{matrix} P(t+h) - P(t+h-n_{am}) \\ h = 1, \ldots, n_{am} \end{matrix}\right\}. \tag{9}$$

The filtered version of the wind profile $P^f(t)$ can also be formulated as a convolution product of the power signal and the difference of boxes edge detector $f_{n_{am}}$, as defined by the relationship [31]

$$f_{n_{am}} = \frac{1}{n_{am}}(\mathbf{1}_{n_{am}} - \mathbf{1}_{n_{am}}) : P^f(t) = P(t) * f_{n_{am}}, \tag{10}$$

where parameter n_{am} represents the number of averaged differences of measures to consider, i.e. the width of the filter $f_{n_{am}}$, thereby characterizing the smoothing parameter. The filtered signal $P^f(t)$

measures how the power signal varies. A ramp is found within an interval when the absolute value of $P^f(t)$ exceeds the threshold P_{val} during the interval, i.e.

$$\left|P^f(t)\right| > \tau\,. \tag{11}$$

Threshold τ is related to criteria such as the weather conditions and terrain complexity. If an interval represents a ramp, the ramp timing and ramp intensity is chosen to correspond to the timing and magnitude of the local maxima of the filtered signal for that interval. In addition, the ramp start and ramp end are defined as the instants where the filtered signal cuts the threshold to increase above and decrease below the threshold, respectively.

Ferreira *et al.* [48] introduced a ramp definition which incorporates a high pass filter, i.e. a filter that passes through signals above a cut-off frequency and attenuates the signals below it. The high pass filter is defined by the relationship

$$y[i] = \alpha(y[i-1] + x[i] - x[i-1])\,, \tag{12}$$

where α is a value in the range [0, 1], and $x[i]$ and $y[i]$ denote the input and output signal at instant i, respectively. When α is near 1, the output will gradually decrease, and the output is significantly influenced by small changes in the input signal. When α is near 0, the output will decrease rapidly. Therefore, large changes in the input, i.e. large values for $(x[i] - x[i-1])$ is required for the output to vary considerably. When the input signal is a constant value, i.e. $(x[i] - x[i-1] = 0)$ it will result in an output decay to zero.

Fig. 3 depicts the implementation of the ramp definition. Furthermore, a band filter may be applied to the output signal y to filter small peaks and only retain values which are greater than a specified threshold, as per the accepted ramp definition.

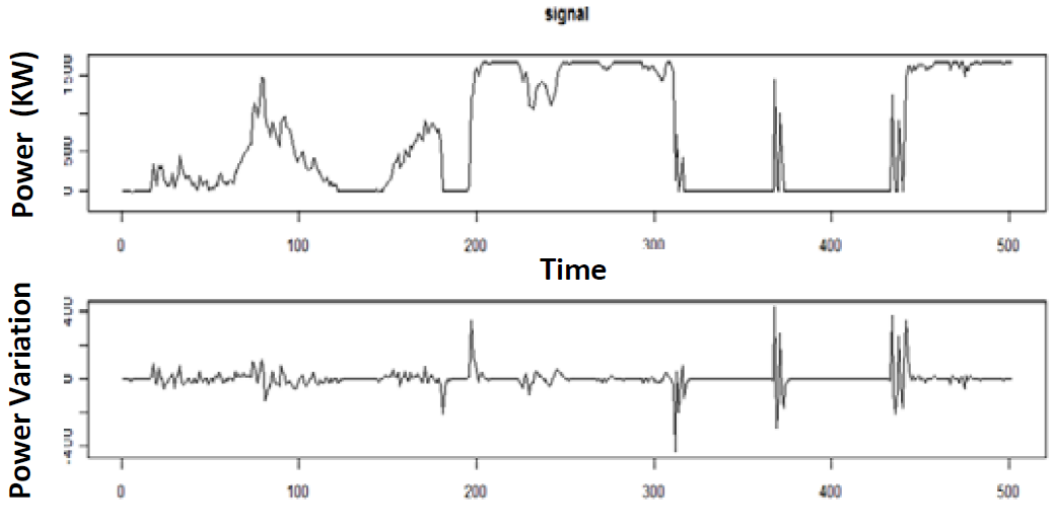


Fig. 3: Top: Wind Power signal. Bottom: High-pass filtered signal with $\alpha = 0.25$ [49].

In conclusion, the ramp definition from (1) is used the most in literature for binary ramp classification [35]. Furthermore, Gallego-Castillo *et al.* [35] summarised the various thresholds of ramp magnitude and ramp duration to perform binary ramp/non-ramp classification found in literature. The wide variation of these

parameters, shown in Table 1, suggests that no consensus regarding thresholds for ramp magnitude and ramp duration has been reached.

Table 1: Binary ramp definitions reviewed in literature [35].

Author	ΔP_0	Δt_r	Case study size	Comment
Cutler [10]	75%P_R	3 h	65 MW	Hourly resolution
Cutler [10]	65%P_R	1 h	65 MW	10-min resolution
Freedman [12]	200 MW	30 min	~1 GW	
Truewind [11]	20%P_R	1 h	n/s (portfolio)	Threshold for ramp-up
Truewind [11]	15%P_R	1 h	n/s (portfolio)	Threshold for ramp-down
Potter [8]	10%P_R	1 h	~1 GW	
Greaves [7]	50%P_R	4 h	3–240 MW	
Barbour [21]	20%P_R	30 min	~200 MW	Core ramp
Collier [25]	50%P_R	4 h	n/s	
Bradford [24]	20%P_R	1 h	n/s (wind farm)	
Bossavy [23]	50%P_R	n/a	n/s (wind farm)	
Kamath [30]	10-12%P_R	30 min	~1 GW	
Kamath [30]	15-20%P_R	1 h	~1 GW	
Gallego-Castillo [34]	σ_g	1 h	33 MW	
Cutler [28]	200 MW	30 min	868 MW	
Cutler [28]	150 MW	5 min	868 MW	Threshold for ramp-down
Cutler [28]	150 MW	30 min	286 MW	
Cutler [28]	75 MW	30 min	140 MW	
Zareipour [31]	50 MW	10 min	n/s (portfolio)	
Pinson [35]	50%P_R	4 h	n/s (wind farm)	
Bossavy [54]	30%P_R	n/a	8 MW	
Yang [56]	15%P_R	1 h	n/a	
Fernandez [42]	25%P_R	3 h	18 MW	
Suzuki [45]	15%P_R	6 h	n/s (wind farm)	
Revheim [58]	30%P_R	3 h	n/a	
Heckenbergerova [57]	50%P_R	5 h	68 MW	
Gan [61]	40%P_R	n/a	n/s (wind farm)	Threshold for ramp-up
Gan [61]	30%P_R	n/a	n/s (wind farm)	Threshold for ramp-down

P_R stands for the rated power of the case study considered. σ_g is the standard deviation of the first-difference of the wind power time series (see text for details). "n/a" and "n/s" stands for "not applicable" and "not specified", respectively.

Accordingly, Gallego-Castillo *et al.* [13], [49] expressed a cause for concern about binary ramp definitions due to drawbacks associated with its use. These drawbacks include the sensitivity of the number of detected ramp events to the specified threshold values, as well as the idea that all ramps are similar [35]. Gallego-Castillo *et al.* [13] therefore proposed a novel methodology, namely the ramp function, to characterise ramp events based on the wavelet transform to address several drawbacks related to ramp event characterisation based on binary definition. A continuous index is provided at each time step of a given wind power time series to represent the associated ramp intensity. It was concluded that characterising the ramp performance via the wavelet-approach is more reliable compared to characterising the ramp performance based on binary ramp definitions. Gallego-Castillo [49] also used the application of the ramp function to characterise ramp events.

2.4.3 Wind power ramp detection algorithms

2.4.3.1 Introduction

Wind power ramp event detection algorithms typically implement a mathematical algorithm to extract these ramp events from a temporal wind power profile according to a user-specified definition of a wind power ramp event. The temporal wind power profile can represent either actual or forecasted wind power data [20]. This section reviews the detection algorithms found in literature, as well as metrics proposed in literature to evaluate and compare the detection performances of various detection algorithms.

2.4.3.2 Detection algorithms

An extensive range of algorithms for detecting wind power ramps have been proposed in literature.

Cutler *et al.* [50] developed a two-stage detection method to identify and categorise large ramp events in wind power output. Wind power ramp events are identified as instances in the wind power output where the wind power changes more than a certain threshold value within a given timeframe. For the first stage, all instances within an hourly averaged wind power profile where a change in power greater than 75% occurred within a timeframe of 1 hour is identified. In addition, if two individual hourly ramps occur less than 6 hours apart, then the ramps are combined. For the second stage, the periods of time that contain wind power ramp events identified by stage 1 are first removed. Thereafter, all instances within the 10-minute averaged wind power profile where a change in power greater than 65% occurred within a timeframe of 1 hour is identified. Similarly, if two individual 10-minute ramps occur less than 1 hour apart, then the ramps are combined. Individual 10-minute ramps which occur on approximately the same day and are also categorised as a variable period are combined. It is anticipated that the approach is highly sensitive to noise because wind power profiles exhibit high variability [40].

Kamath [14], [37] proposed a simplified detection technique to identify all wind power ramp events in a wind power time series with durations of 5, 15, 30 and 60 min, respectively. The technique entails testing each time series sample to determine if the ramp definition is met under a given threshold duration value. The detection technique performs satisfactorily when the sampling interval of the wind power time series is large, i.e. larger time scales, and when the wind power times series is sufficiently filtered. The drawback of this technique is, however, that it overcounts ramps that have a large duration [21]. In a similar way, Taylor [42] detected a ramp when the change in wind power, in the hourly wind power data, exceeded a user-specified threshold value. The above-mentioned methods are simple, however, it is not capable of detecting all the ramps of varying durations or rates present in the wind power time series that satisfy the user-specified ramp definitions [21].

Recently, the Swinging Door Algorithm (SDA), an algorithm initially proposed by Bristol [51] for data compression, has been employed in the renewable energy community to extract ramps. Florita *et al.* [16] proposed the application of the SDA to extract ramp intervals in a piecewise linear fashion from wind and solar power data. The ramps are regarded as segments in which the wind power is considered to follow a single trend. The methodology incorporates a tunable parameter, epsilon, that influences the sensitivity to ramp variations. A user specified definition of a significant ramp is applied to the ramp intervals obtained with the method to detect the wind power ramp events. The benefits of the SDA include its structural and computation efficiency as well as its robustness, regardless of noise [16].

Ouyang *et al.* [52] proposed an improved swinging door algorithm, based on the SDA proposed by Florita *et al.* [16], to extract ramp events from historical data. The same concept of the swinging door algorithm is used to extract segments as linear trends, however, the implementation employs the use of a parallelogram instead of the swinging doors. The algorithm is explained only in a descriptive way. Ouyang *et al.* [53] subsequently proposed an improved SDA to conveniently extract linear segments from a predicted wind power signal. Although the same descriptive explanation from Ouyang *et al.* [52] was employed, an explicit mathematical formulation of the algorithm was also included by Ouyang *et al.* [53]. Despite the benefits of the SDA, it still has several issues associated with its use, including that there is no consensus on how to determine the optimal value of the tuneable parameter and how the optimal ramp segments for the wind power time series should look [34], [54]. The start- and end-points of the detected

ramps are, furthermore, not always accurate. In a related methodology, Makarov *et al*. [33] applied a swinging window algorithm to derive the required ramping capability, ramping rate and ramping duration for each wind power interval.

Sevlian and Rajagopal [21], [55] proposed the L1-ramp detect with Sliding Window (L1-SW) methodology to identify wind power ramp events in a large time series. L1-trend filtering is applied to the data as a preprocessing step to remove noise that is not within the appropriate time scale and to emphasise appropriate trends in the data. The method uses a family of scoring functions that are related to a user defined ramp rule. A dynamic programming recursion technique is then used to extract all significant ramp events. The method is advantageous because it always finds the largest intervals within the wind power time series that satisfies the user-specified ramp rules, and it also ensures that all ramps satisfying the given rules are detected [21]. Additionally, the method facilitates the use of various ramp rules.

Zhang *et al.* [30] developed an Optimised Swinging Door Algorithm (OpSDA) to improve the performance of the SDA by reducing the number of linear ramp segments. The method entails that the wind power time series is first segmented using the original SDA [16], after which the significant ramps are extracted. A dynamic programming algorithm is applied, adopted from Sevlian and Rajagopal [21], [55], to merge the significant ramps with the same direction. It is concluded that the OpSDA performs significantly better compared to the original SDA, as well as similarly or better compared to the L1-SW, while also being more computationally inexpensive. It is recognised that the ramps detected with the SDA and the OpSDA are anchored to the measurement points of the wind power signal, whereas the ramps detected with the L1-SW is anchored to the points of the filtered signal. Furthermore, a framework was developed to determine the optimal value of the tuneable parameter in the SDA using the OpSDA as a baseline, since in some cases it may be preferable to employ the SDA because of its computational expedience.

Cui *et al.* [54] subsequently improved the OpSDA to merge significant adjacent ramps with the same direction, and manage bumps and insignificant ramp intervals. Cui *et al.* [20] proposed a further improved OpSDA to extract ramping events in wind power, solar power, load and netload. The OpSDA was improved for wind power ramps by merging bumps into adjacent ramping segments with the same direction, however, it did not include the processing of insignificant ramp intervals. Additionally, another implementation of the SDA was used to segment the wind power time series, which is similar to the swinging window algorithm adopted by Makarov *et al*. [33].

The various detection algorithms have been adopted in literature for a wide range of applications. Zhang *et al.* [19] adopted the SDA from Florita *et al.* [16] to extract ramp events from actual and forecasted wind power data to evaluate the performance of wind power ramp forecasts based on improved short-term wind power forecasts. Zhang *et al.* [34] adopted the OpSDA from Zhang *et al.* [30] to identify all the ramp events in actual and forecasted wind power data to evaluate the performance of wind power ramp forecasts based on improved short-term wind power forecasts. Cui *et al.* [45] used the OpSDA from Zhang *et al.* [30] to extract all wind power ramps and their corresponding features from historical wind power data in order to characterise the probability distributions of the different ramping features via a Generalised Gaussian Mixture Model (GGMM). Cui *et al.* [56] adopted the OpSDA from Zhang *et al.* [30] to extract wind power ramps from historical data in order to construct a database of historical ramping features. Cui *et al.* [57] applied the optimised swinging door algorithm, adopted from Zhang *et al.* [30], to an ensemble of wind power forecasting scenarios to extract all wind power ramps so to develop a probabilistic wind power ramp forecast.

2.4.3.3 Performance metrics for comparison of ramp detection algorithms

Sevlian and Rajagopal [21] evaluated the accuracy of ramp detection by comparing the ramps detected by the L1-SW to manually detected ramps for a small portion of the wind power data. Ramps are detected manually by visually inspecting the wind power profiles and identifying instants where large and rapid changes in wind power are observed and recording the start- and end-points that demarcate the observed ramps. The manually detected ramps and the algorithm detected ramps are compared to identify equivalent ramps, i.e. ramps detected by both methods, as well as the ramps that are detected by one method but missed by the other method. Two ramps are considered to be equivalent if the length of the overlapping interval exceeds a certain threshold, which is chosen as 80% of the mean length of the two ramps. Furthermore, an accuracy matrix is constructed based on the comparison of the sets of ramps, as summarised in Table 2. The accuracy matrix includes the number of manually detected ramps that are accurately detected by the L1-SW. It also includes the number of manually detected ramps that are not detected by the L1-SW, as well as the number of ramps detected by the L1-SW that are not manually detected. The accuracy matrix provides a measure of the skill of the L1-SW algorithm compared to manual detection.

Table 2: Accuracy of ramp detection by comparing manually detected ramps and algorithm detected ramps [21].

	Manual	**Manual Not Detected**
Algorithm Detected		
Algorithm Not Detected		

It is crucial that system operators have accurate information about when ramps start and end, as this information is useful in various power system applications and in making scheduling decisions to minimise the cost associated with ramp events [20]. Accordingly, Cui et al. [54] proposed a set of metrics to evaluate and compare the ramp detection performance of the OpSDA and the L1-SW. The set of metrics is given by

$$[p(t+1) - p(t)] \times [p(t) - p(t-1)] < 0 \qquad (13)$$

and

$$[p(t+1) - p(t)] \times [p(t) - p(t-1)] > 0\,, \qquad (14)$$

where t indicates the current time and $p(t)$ denotes the power generation at time t. The metrics in (13) and (14) are used to determine whether the start- or end-point of a given Wind Power Ramp Event (WPRE) is extracted successfully or not. When the criteria in (13) is met for the start- or end-point of a given WPRE, an inflection point occurred at the point and therefore the point is successfully extracted. Conversely, when the criteria in (14) is met for the start- or end-point of a given WPRE, an inflection point did not occur at the point and therefore the point is not extracted successfully. The metrics should be tested for the start-points and end-points of all the WPREs. The statistical results of the metrics are summarised in a contingency table. Additionally, Cui et al. [54] compared the OpSDA and L1-SW based on visual inspection of the detected ramps, statistical analysis of the key ramp characteristics and computation time.

Cui *et al.* [20] proposed a similar set of metrics to evaluate the ramp detection performance of load ramp events, netload ramp events, solar power ramp events and wind power ramp events, based on the detection accuracy of the start- and end-points of the ramps. The proposed metrics are similar to the

metrics proposed by Cui et al. [54] as both are based on the same concept, specifically, a ramp start or ramp end is accurately detected when it corresponds to an inflection point , i.e. a local maxima or minima, otherwise the ramp start or ramp end is inaccurately detected. However, the proposed metrics are formulated slightly different than (13) and (14). The proposed metrics is given by

$$\frac{[p(i+DT)-p(i)]}{[p(i)-p(i-DT)]} < 0, DT \in \{1-\ ,5-\ ,15-, and\ 60-minute(s)\}\ (YES) \qquad (15)$$

and

$$\frac{[p(i+DT)-p(i)]}{[p(i)-p(i-DT)]} > 0, \qquad DT \in \{1-\ ,5-\ ,15-, and\ 60-minute(s)\}\ (NO)\ , \qquad (16)$$

where i represents the current time, DT represents the time resolution of the sampling data and $p(i)$ denotes the power generation at time instant i. The metric listed in (15) corresponds to that of (13) and the metric listed in (16) corresponds to that of (14).

Cui *et al.* [20] presented a more detailed description of the concept on which the metrics are based and included illustrations as shown in Fig. 4 to Fig. 7. Fig. 4 depicts an accurate end of an upward ramp or accurate start of a downward ramp, and Fig. 6 depicts an accurate end of a downward ramp or accurate start of an upward ramp. Conversely, Fig. 5 depicts an inaccurate start or end of an upward ramp, and Fig. 7 depicts an inaccurate start or end of a downward ramp.

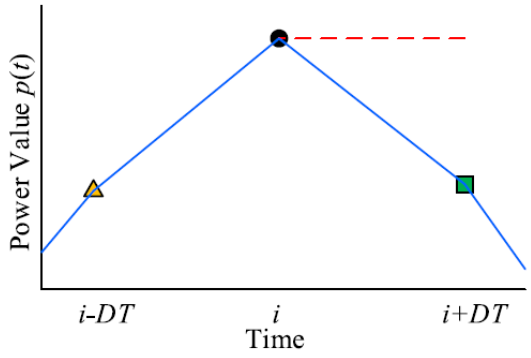


Fig. 4: Accurately detected (YES) the end of an upward ramp or start of a downward ramp [20].

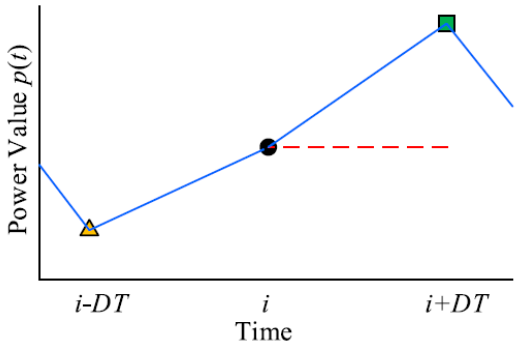


Fig. 5: Inaccurately detected (No) the start or end of an upward ramp [20].

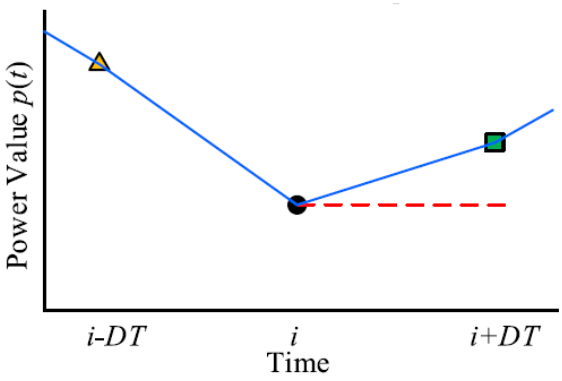


Fig. 6: Accurately detected (YES) the end of a downward ramp or start of an upward ramp [20].

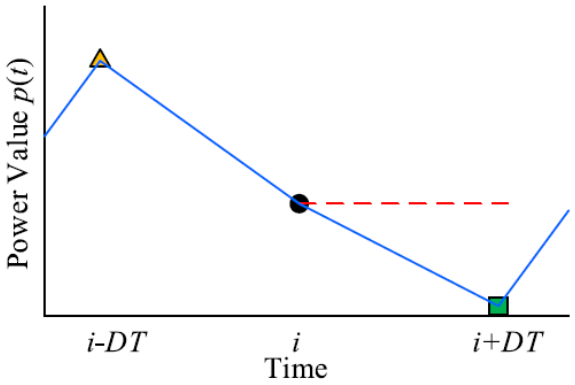


Fig. 7: Inaccurately detected (No) the start or end of a downward ramp [20].

The ramp detection evaluation results are used to construct a contingency table, as shown in Table 3, and a performance diagram for the purpose of statistical and visual analysis. For the contingency table, each ramp within the set of detected ramps is assigned to one of four groups based on the accuracy of its ramp start and ramp end. The four groups include start-Yes-end-Yes (sYeY), start-Yes-end-No (sYeN), start-No-end-Yes (sNeY) and start-No-end-No (sNeN). Correspondingly, the number of ramps with an accurate start- and end-point is denoted by sYeY. The number of detected ramps with accurate start-points, but inaccurate end-points, are represented by sYeN. The number of detected ramps with inaccurate start-points, but accurate end-points, are represented by sNeY. The number of detected ramps with inaccurate start- and end-points are denoted by sNeN [20]. The same contingency table is employed by *Cui et al.* [54], although the development of the contingency table is not explained.

Table 3: Contingency table for evaluating the ramp detection performance based on the detection accuracy of the start- and end-points of the ramps [20].

	End (Yes)	**End (No)**
Start (Yes)	sYeY	sYeN
Start (No)	sNeY	sNeN

Cui et al. [20], furthermore, proposed a suite of metrics, derived based on the contingency table to evaluate the ramp detection performance, namely Probability of Detection (POD), Critical Success Index (CSI), Frequency Bias Score (FBS) and Success Ratio (SR). These metrics are defined by the relationships

$$POD = \frac{sYeY}{sYeY + sNeY}, \tag{17}$$

$$SR = \frac{sYeY}{sYeY + sYeN}, \tag{18}$$

$$CSI = \frac{sYeY}{sYeY + sNeY + sYeN} \tag{19}$$

and

$$FBS = \frac{sYeY + sYeN}{sYeY + sNeY}. \tag{20}$$

POD represents the fraction of the detected ramp events with accurate end-points that have accurate start-points. SR represents the fraction of the detected ramp events with accurate start-points that have accurate end-points. CSI represents the fraction of ramp events with an accurate ramp start or an accurate ramp end that have both an accurate ramp start and ramp end. The CSI ranges from 0 to 1, where 0 represents the worst detection accuracy and 1 represents the best detection accuracy. The FBS represents the ratio of the ramp events with accurate start-points to the ramp events with accurate end-points. If the FBS has a value smaller than one, the number of ramp events with accurate end-points exceeds the number of ramp events with accurate start-points. Conversely, if the FBS is greater than one, the number of ramp events with accurate start-points exceeds the number of ramp events with accurate-end-points.

The relationship between the suite of metrics is defined by the relationships

$$CSI = \frac{1}{\frac{1}{POD} + \frac{1}{SR} + 1} \tag{21}$$

and

$$FBS = \frac{POD}{1 - FAR} = \frac{POD}{SR}, \tag{22}$$

where FAR represents the False Alarm Ratio. The definition for the FAR was not included by Cui et al. [20], however, it can be derived from Table 3, using the definition of the FAR employed by Zhang *et al.* [34] as reference. Zhang *et al.* [34] employs the POD, CSI, FBS, SR and FAR to evaluate the ramp forecasting performance. The FAR, as derived from Table 3, is given by the formulation

$$FAR = \frac{sYeN}{sYeY + sYeN}. \tag{23}$$

The relationship between the suite of metrics can be visualised on a performance diagram.

Cui et al. [30], [54] developed a framework to determine the optimal value of the tunable parameter, ε, in the SDA based on using the optimised SDA as a baseline. In other words, the tunable parameter of the SDA is optimal when the ramps detected by the SDA for the optimal tunable parameter approaches the ramps detected by the OpSDA. Therefore, a contingency table similar to the one shown in Table 2 is employed to measure the skill of the SDA approaching that of the OpSDA, as shown in Table 4. In this contingency table, *True Positive* (TP) represents the number of ramps that are detected by the SDA and OpSDA. *False Negative* (FN) represents the ramps that are detected by the OpSDA but not by the SDA. *False Positive (FP)* denotes the ramps that are detected by the SDA but not by the OpSDA. *True negative* (TN) represents the number of non-ramp events detected by both.

Table 4: Contingency table for the SDA and OpSDA [30], [54].

	OpSDA (YES)	**OpSDA (No)**	**Total**
SDA (YES)	TP (hits)	FP (false alarm)	TP+FP
SDA (No)	FN (misses)	TN	FN+TN
Total	TP+FN	FP+TN	N=TP+FP+FN+TN

A suite of metrics is derived from the contingence table to evaluate the ramp detection performance of the SDA for various ε values using the OpSDA as baseline. The suite of metrics include the POD, CSI, FBS and SR [19], [20], [34]. These metrics are employed by Zhang *et al.* [19], [34] to evaluate the ramp forecasting performance and are used by Cui *et al.* [20] to evaluate the ramp detection performance based on the detection accuracy of the ramp start- and end-points, as mentioned previously. It is recognised that since Cui *et al.* [20] and Cui et al. [30], [54] employ the suite of metrics for different applications, accordingly, the elements from the respective contingency tables also differ. The suite of metrics derived from the contingency table shown in Table 4 is defined by the relationships

$$POD = \frac{TP}{TP + FN}, \tag{24}$$

$$SR = \frac{FP}{FP + TP}, \tag{25}$$

$$CSI = \frac{TP}{TP + FN + FP} \tag{26}$$

and

$$FBIAS = \frac{TP + FP}{TP + FN}. \tag{27}$$

POD represents the fraction of the ramp events detected by the OpSDA that are also detected by the SDA. SR represents the fraction of the ramp events detected by the SDA that are not detected by the OpSDA. CSI represents the fraction of ramp events detected by the SDA or the OpSDA that are detected by both methods. The CSI ranges from 0 to 1, where 0 represents the worst possible correspondence and 1 represents the best correspondence The FBS represents the ratio of the ramp events detected by the SDA to the ramp events detected by the OpSDA. If the FBS has a value smaller than one, the number of ramp events detected by the OpSDA exceeds the number of ramp events detected by the SDA. Conversely, if the FBS is greater than one, the number of ramp events detected by the SDA exceeds the number of ramp events detected by the OpSDA. In conclusion, the optimal value of the tunable parameter is determined according to the largest POD.

2.5 Regression analysis

2.5.1 Overview

Regression analysis is a statistical technique that is used to explore and model the statistical relationships between variables, specifically a response variable and one or more regressor variables [58], [59], [60], [61]. Regression analysis typically employs a mathematical equation to estimate the statistical relationship between the variables of interest [58], [59], [61], namely response variables and regressor variables. The response variable refers to the variable of interest that is to be predicted or modelled, while the regressor variables are the variables that explain how the response variable changes [60], [62]. The technique can also be applied to derive inferences about a larger data set, or to predict values of the response variable given values of the regressor variables [63].

Regression analysis is an iterative procedure, where the data is firstly studied to develop a preliminary model. The model is subsequently fitted to the data, and the quality and appropriateness of the fit is

assessed. Accordingly, the model is either adopted or revised until the model is deemed adequate [58], [59]. It is important to note that the relationship between the variables, approximated by the regression equation, is only valid for the range of observed regressor variables [58]. The regression models do not signify causality for the variables of interest [58], [64].

2.5.2 Simple linear regression model

A simple linear regression model incorporates only one regressor variable, and describes the relationship between the response variable y and the regressor variable x using a straight-line [58], [64]. The simple linear regression model is linear in its parameters as well as linear in the regressor variable [58], [59]. The simple linear regression model can, furthermore, be used to represent nonlinear relationships between the response and regressor variable by transforming the regressor variable [58], [59], [60]. The population regression model is defined by the relationship

$$y = \beta_0 + \beta_1 x + \varepsilon\,, \tag{28}$$

where β_0 and β_1 represent unknown model parameters that are estimated from the sample data. Parameters β_0 and β_1 are generally called the regression coefficients and represent the intercept and slope of the regression line, respectively [58]. Alternatively, (28) can be expressed as the sample regression model, written in terms of n pairs of observations (x_i, y_i), $(i = 1,\ 2, \ldots, n)$, where

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, i = 1, 2, \ldots, n\,. \tag{29}$$

The slope defines the amount the y value changes by when x increases by one unit [58], [62]. When the range of x includes 0, the intercept is the value of y when $x = 0$. When the range of x excludes 0, however, the intercept does not have a practical meaning. Furthermore, ε represents a random error component which accounts for the observations that do not fall exactly on the regression line due to fundamental randomness or shortcomings in the model. Therefore, ε represents the difference between the regression line and the y-values of the observations.

It is assumed that the errors are uncorrelated, with a mean of zero and an unknown, constant variance of σ^2 [58], [64]. Furthermore, an additional assumption is sometimes made about the form of the probability distribution of ε, namely that errors are normally distributed. This assumption is made in order to facilitate inference and hypothesis tests and to improve the estimation performance [64]. In this case, the errors are also independent in the normal error model [59].

It is recognised that the regressor variable x is a controlled variable, whereas the response variable y is a random variable [58], [59]. In other words, for each possible value of the regressor variable x, the response variable y has a probability distribution. Based on the assumptions of the error term, the probability distribution of y has a mean of $E(y|x)$ and variance of $Var(y|x)$, as formulated by the expressions [58], [62], [59]

$$E(y|x) = \beta_0 + \beta_1 x \tag{30}$$

and

$$Var(y|x) = Var(\beta_0 + \beta_1 x) = \sigma^2\,. \tag{31}$$

From (30), the mean value of y is a linear function of x [58]. Correspondingly, the regression function for the simple linear regression model is defined by (30) [59]. From (31), it is assumed that the variance of y is constant σ^2 and independent from x [58], [59].

The simple linear regression model is completely described when the model parameters β_0, β_1 and σ^2 are known.

2.5.3 Least squares estimation of the regression parameters

Parameters β_0 and β_1 are initially unknown. Various methods can be used to estimate these parameters from the given data, including the Ordinary Least Squares (OLS) method, Generalised Least Squares method, Weighted Least Squares method, Maximum Likelihood method, Restricted Least squares method, Ridge Regression method and Random Coefficient method. The OLS method will be considered in this section to fit the regression line and estimate the unknown parameters β_0 and β_1 [58], [64].

The objective of the Least Squares method is to minimise the sum of squares of the difference between the observed values y_i and the regression line. Accordingly, the least-squares criterion to be minimised is given by

$$S(\beta_0, \beta_1) = \sum_{i=i}^{n} (y_i - \beta_0 - \beta_1 x_i)^2 \,. \tag{32}$$

The least squares estimator of the intercept, $\widehat{\beta_0}$, and the least squares estimator of the slope, $\widehat{\beta_1}$, is given by

$$\widehat{\beta_0} \;=\; \bar{y} - \widehat{\beta_1}\bar{x} \tag{33}$$

and

$$\widehat{\beta_1} = \frac{\sum_{i=1}^{n} y_i x_i - \frac{(\sum_{i=1}^{n} y_i)(\sum_{i=1}^{n} x_i)}{n}}{\sum_{i=1}^{n} {x_i}^2 - \frac{(\sum_{i=1}^{n} x_i)^2}{n}}, \tag{34}$$

respectively. The sample means $\bar{x}$ and $\bar{y}$ are given by

$$\bar{y} = \frac{1}{n}\sum_{i}^{n} y_i \tag{35}$$

and

$$\bar{x} = \frac{1}{n}\sum_{i}^{n} x_i \;, \tag{36}$$

respectively. It follows that the estimated regression function that defines the fitted value $\hat{y}$ as a function of x and the estimated model parameters $\widehat{\beta_0}$ and $\widehat{\beta_1}$ is given by

$$\hat{y} = \widehat{\beta_0} + \widehat{\beta_1}\, x \,. \tag{37}$$

Essentially, $\hat{y}$ is a point estimate of the mean of y for a given x [59].

A residual, denoted by e_i, is defined as the difference between the observed value y_i and the corresponding fitted value $\hat{y}_i$. This yields

$$e_i = y_i - \hat{y}_i = y_i - \widehat{\beta_0} + \widehat{\beta_1}\, x_i, i = 1, 2, \dots, n\,. \tag{38}$$

It is necessary to make a distinction between the model error term ε_i and the residual e_i. The model error term is unknown as it involves the difference between the unknown regression line and the observed y values. Conversely, the residual is known, from its above-mentioned definition [59].

Following the implementation of the least squares method, it is necessary to investigate the adequacy of the model, which results in either the adoption of the model or the modification of the model [58]. The residuals are instrumental in determining whether the model is adequate or not, by checking whether the underlying model assumptions are satisfied or not [58],[59].

2.5.4 Properties of the least squares estimators and the fitted regression model

2.5.4.1 Properties of the least squares estimators

The estimated model parameters have various important properties, including the following:

- The estimated model parameters $\widehat{\beta_0}$ and $\widehat{\beta_1}$ are linear combinations of the observed values y_i [58], [59].
- The estimated model parameters $\widehat{\beta_0}$ and $\widehat{\beta_1}$ are unbiased estimates of β_0 and β_1, i.e.

$$E\{\beta_0\} = \widehat{\beta_0} \tag{39}$$

and

$$E\{\beta_1\} = \widehat{\beta_1}\,, \tag{40}$$

regardless of the form of the distribution of the error terms [58], [59].

- The estimated model parameters are the most precise estimators, i.e. the estimated model parameters have minimum variance, among all unbiased linear estimators [58], [59].

2.5.4.2 Properties of the fitted regression line

The regression line fitted by the least squares method has several important properties, including the following [58], [59]:

- For any regression model that includes an intercept β_0, the sum of the residuals equals 0, i.e.

$$\sum_{i=1}^{n} (y_i - \hat{y}_i) = \sum_{i=1}^{n} e_i = 0\,. \tag{41}$$

- The sum of the observed values y_i is equivalent to the sum of the fitted values $\hat{y}_i$, i.e.

$$\sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \hat{y}_i\,. \tag{42}$$

- The regression line always crosses the centroid of the data given by $(\bar{x}, \bar{y})$.
- The weighted sum of the residuals equals 0 when the given weights are the corresponding regressor variable, i.e.

$$\sum_{i=1}^{n} x_i e_i \,. \tag{43}$$

- The weighted sum of the residuals equals 0 when the given weights are the corresponding fitted value, i.e.

$$\sum_{i=1}^{n} \hat{y}_i e_i \,. \tag{44}$$

2.5.5 Estimation of standard deviation of residuals

An estimate of the constant variance σ^2 of each residual e_i, or each observation y_i for the regression model, is obtained via the residuals sum or squares, otherwise referred to as the error sum of squares, SS_{Res}, given by [58]

$$SS_{Res} = \sum_{i=1}^{n} {e_i}^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \,. \tag{45}$$

The residual sum of squares has $n - 2$ degrees of freedom, as the estimated model parameters $\widehat{\beta_0}$ and $\widehat{\beta_1}$ account for two degrees of freedom. Consequently, an unbiased estimator of σ^2 is given by

$$\hat{\sigma}^2 = \frac{SS_{Res}}{n-2} = MS_{Res} \,, \tag{46}$$

where MS_{Res} is called the residual mean square [58], [59]. Furthermore, an estimate of the standard deviation σ for the regression model is the positive square root of $\hat{\sigma}^2$ [59]. The square root of $\hat{\sigma}^2$ is generally referred to as the standard error of regression, and its units correspond to that of the response variable y [58]. The standard error of regression measures the typical difference between the observed value y_i and its corresponding fitted value $\hat{y}_i$.

The usefulness of $\hat{\sigma}^2$ as an estimate of σ^2 greatly depends on the compliance of the model errors to the underlying assumptions and the appropriateness of the model, as $\hat{\sigma}^2$ depends on the residuals [58].

2.5.6 Key goodness-of-fit measures

The coefficient of determination is a descriptive measure used to describe the degree of linear association between x and y [59], [65]. The coefficient of determination is given by the relationship [58]

$$R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_{Res}}{SS_T}, \tag{47}$$

where

$$SS_T = \sum_{i=1}^{n} (y_i - \bar{y})^2 \tag{48}$$

and

$$SS_R = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 \,. \tag{49}$$

SS_T measures the total variability in y, i.e. without taking the regressor variable into account. SS_R is generally called the regression or model sum of squares and measures the variability in y which is explained by the simple linear regression model. SS_{Res} is the residual or error sum of square, as previously formulated in (45), that represents the residual variability in y which is not explained by the simple linear regression model. Consequently, the coefficient of determination is the fraction of the total variability in y that is explained by the simple linear regression model, specifically the variability in y according to the linear relationship [58], [65]. It is recognised that the coefficient of determination is equivalent to the square of the Pearson correlation coefficient between x and y [64], [65].

The value of R^2 ranges from 0 to 1, since $0 \leq SS_{Res} \leq SS_T$ [58], [64]. If $R^2 = 1$, the observations and the regression line match [59]. If R^2 is close to 1, a large degree of the total variability in y is explained by the regression model [58]. If $R^2 = 0$, it is recognised that $\hat{y}_i = \bar{y}$ and the regression line is horizontal, i.e. $\beta_1 = 0$. Therefore, if $R^2 = 0$, there is no linear relationship between y and x [59].

It is necessary to use R^2 cautiously, since the value of R^2 may increase when enough terms are added to the regression model and since the R^2 does not decrease when terms are added [58]. Nevertheless, a higher R^2 will not necessarily imply a better model. Furthermore, the value of R^2 is sensitive to the range of variability in the regressor variable [58].

Several misconceptions are associated with the coefficient of determination, including the following [58], [59]:

- The coefficient of determine is not related to the magnitude of the slope of the regression line.
- The coefficient of determination does not measure whether a model is appropriate or not.
- A large coefficient of determination does not signify that the regression model will be able to make useful predictions, as no information is provided on the precision of the model for estimation.
- A coefficient of determination value of 0 does not indicate that no relationship exists between x and y.

2.5.7 Statistical hypothesis test

It is commonly considered beneficial to test hypothesis about the model parameters and to construct confidence intervals [58]. It is necessary to assume normal error regression model for these procedures [58]. Specifically, making conclusions about the slope of the regression line β_1 is deemed important, which necessitates tests involving β_1 [59].

One important hypothesis test involves testing whether the slope of the regression line equals 0, i.e. $\beta_1 = 0$. This test is considered important because when $\beta_1 = 0$ it implies that there is no linear relationship between y and x and the regression line is horizontal, as well as the means of the probability distributions of y are all the same. Furthermore, if the errors are normally distributed and $\beta_1 = 0$, it implies that there is no relationship whatsoever between y and x. Two approaches can be used to test significance of regression, namely t test of Analysis-of-Variance (ANOVA) F-test [58].

2.5.8 Model adequacy checking

There are several model assumptions that have to be satisfied for the OLS model to ensure its validity and adequacy [63]. If the model assumptions are not satisfied for the OLS model, it may result in biased predictions or estimation on the model [65]. In short, it is assumed that the relationship between the response variable y and the regressor variable x is linear, and that the error terms are uncorrelated with a mean of 0 and a constant variance of σ^2. Furthermore, it is required to assume that the errors are normally distributed for hypothesis testing and interval estimation, and when the errors are both normally distributed and uncorrelated, the errors are independent random variables.

Residual analysis is useful to determine whether the underlying model assumptions are satisfied, in particular for the simple linear regression model with normally distributed errors [58]. Since the residuals are viewed as the observed model errors, the residuals are required to imitate the underlying assumptions of the model errors [58].

It may sometimes be preferable to use standardised residuals. Standardised residuals can help to discover outliers in the data, i.e. observations that are significantly different from the rest of the data [58], [59]. The standardisation technique is defined by the commonly used expression [58], [59]

$$e_i^* = \frac{e_i - \bar{e}}{\sqrt{MS_{Res}}} = \frac{e_i}{\sqrt{MS_{Res}}} \qquad (50)$$

with

$$\bar{e} = \frac{1}{n}\sum_i^n e_i = 0\,, \qquad (51)$$

where e_i^* is called a standardised residual and has a mean of zero and variance that is approximately equal to 1, and $\bar{e}$ is the mean of the residuals [58].

Graphical analysis of the residuals or formal statistical test can be used to perform residual analysis to investigate the model adequacy and check the underlying model assumptions [59]. The basic residual plots include the following:

- Plot of residuals versus regressor variable.
- Plot of absolute or squared residuals versus predictor variable.
- Plot of residuals versus fitted values $\hat{y}_i$.
- Plot of residuals versus time sequence or outlier sequence.
- Plots of residuals versus omitted predictor variables.
- Box plot of residuals.
- Normal probability plot of residuals.

The statistical tests may be applied to the residuals to obtain quantitative measures of some of the model inadequacies, since the diagnostic plots are subjective [58], [59]. Diagnostic plots are, however, still preferable. Most of the statistical tests require that the errors are normally distributed [59]. A run test or the Durbin-Watson test can be used to test for randomness in residuals. The Brown-Forsythe test and the Breusch-Pagan test can be used to test for constancy of variance. The chi-square test, the Kolmogorov-Smirnov test or the Lilliefors test can be used to test for normality of residuals.

2.5.9 Detection and treatment of outliers

An important step in regression analysis is to detect the outliers and influential observations [61]. Outliers are extreme observations, i.e. the observations that are significantly different from the rest of the data [58], [59]. Two types of outliers can occur in the data. The first type are those outliers with extreme regressor values, i.e. the x-location is far from the mean x value, which is commonly referred to as high-leverage points. The second type are those outliers whose y-value differs greatly from the regression line, which is commonly referred to as outliers [60]. Outliers can significantly affect the least squares fit, depending on the x-location of the observation and the influence it exerts [58], [59].

It is necessary to investigate outliers in order to explain its occurrence, so that the outliers can be appropriately dealt with. The outliers may result from faulty data collection or the outliers may be acceptable. Outliers that result from faulty collection can negatively affect the least-squares regression fit because the regression line is pulled toward the outlier in order to minimise least squares criterion. Consequently, it is desirable to correct these outliers or in some cases to discard the outlier. However, it is important to confirm whether the outlier is indeed a mistake with nonstatistical evidence before the outlier is corrected or discarded. Conversely, an acceptable outlier needs to be retained since it provides valuable information about the key model properties and the inherent variability of the data [58] , [59].

Several methods can be used to detect outliers in linear regression [61]. Potential outliers can be identified by investigating the magnitude of the residuals corresponding to all observations. When the magnitude of the residual value for an observation is significantly larger than the rest, i.e. three or four standard deviations from the mean, a potential outlier is identified [58], [59]. Various plots can be used to identify outliers in the data, namely the residual versus fitted value or regressor value plot, the normal probability plot, box plots, stem-and-leaf plots, and dot plots of the residuals [58], [59]. The scaled residuals also prove to be helpful [58], [59]. Additionally, influence measures can also be used to identify outliers, namely Cook's distance, $DFFIT_i$ and $DFBETA_i$. These measures are based on examining how the deletion of each observation will affect the estimated coefficients, the predicted values and the residuals. Cooks' distance measures how deleting the i^{th} observation will affect all the fitted values. $DFFIT_i$ measures how deleting the i^{th} observation will affect a predicted value y_i. $DFBETA_i$ measures how deleting the i^{th} observation will affect the estimates of regression coefficients. Furthermore, the covariance matrix or hat matrix are also possible detection methods [59].

It may be preferable to use an alternative estimation method that does not give priority to the outliers. Since, the presence of outliers in the data may distort the estimated regression line when employing least-squares and maximum likelihood estimators [59].

2.6 Particle swarm optimisation

2.6.1 Overview

Particle swarm optimisation (PSO), originally introduced by Kennedy and Eberhart [66], is a stochastic optimisation algorithm based on a model of swarm intelligence [67]. It is inspired by the social behaviour of animals, including birds, fishes and insects. The swarm work together to find food and the movement of each member during the search depends on its own learning experiences as well as the learning experiences from other members [67].

The objective of an optimisation problem is to find the best solution in the search space. PSO utilises numerous particles that are randomly placed in the search space of a given optimisation problem to iteratively search for the best solution. The search space represents the set of all the possible solutions of the optimisation problem and every particle represents one possible solution of the optimisation problem [67]. The performance of the particle in solving the optimisation problem is evaluated via a pre-defined objective function [68]. Accordingly, the value of the objective function is evaluated at the current position of each particle. The movement of each particle through the search space is determined according to a combination of factors, namely the particle's current position, its best previous position, the best position from all the particles as well as random perturbations [67]. The movement of the particles occurs iteratively, i.e. a new iteration will start after the position of each particle is revised. Eventually, all the particles in the swarm will converge to an optimal solution of the optimisation problem [67].

Each particle is associated with three D-dimensional vectors, namely the current positions X_i, best position P_i and velocity V_i, The current position X_i is a set of coordinates that represents a point in the search space, and is iteratively updated as the particle moves through the search space trying to find the optimal solution. The current position X_i represents a single solution that corresponds to the evaluation of the objective function at that point. The best position P_i is the set of coordinates that is associated with the best solution that the i^{th} particle has achieved so far. The value of the objective function associated with P_i is stored in the variable $pbest_i$ to facilitate the comparison between the possible solutions in the search space. The velocity V_i represents the movement that the i^{th} particle undergoes during each iteration to its new position, and the velocity V_i is also adjusted at each iteration [67].

The global best P_g represents the coordinates of the best solution that any particle has achieved so far. The value of the objective function at P_g is given by $gbest_i$ [67], [69]. The optimal solution of the optimisation problem is given by P_g and $gbest_i$, if the ending condition is satisfied.

The implementation of the particle swarm optimisation is illustrated in Fig. 8 and Fig. 9.

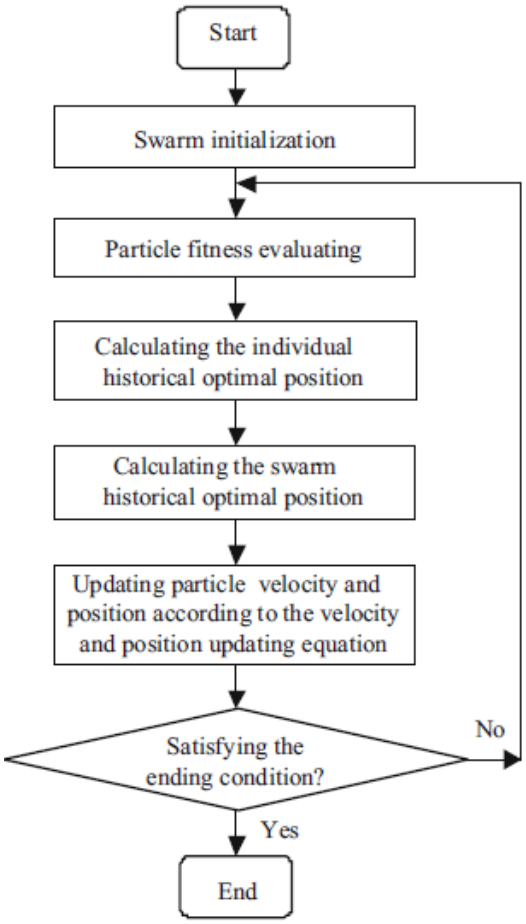


Fig. 8: Workflow of the particle swarm optimisation algorithm [70].

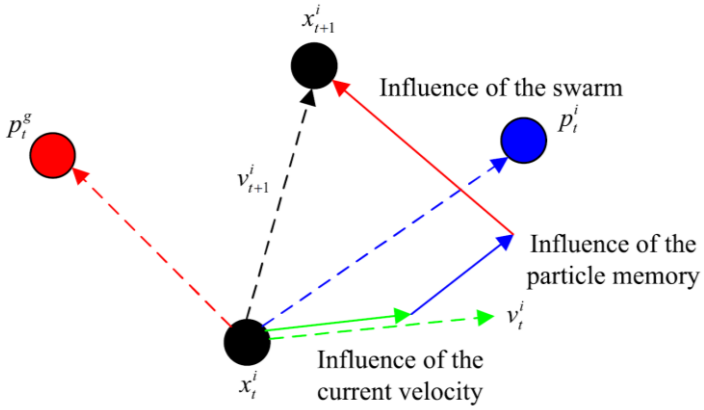


Fig. 9: Iteration procedure of the i^{th} particle in particle swarm optimisation [70].

2.6.2 Particle swarm optimisation in a real number space

The PSO can be described mathematically as follows, given that the size of the swarm is N and D represents the dimensionality of the search space. The position vector, X_i, and velocity vector , V_i, of the i^{th} particle is given by [70]

$$X_i = (x_{i1}, x_{i2}, \ldots, x_{id,}, \ldots, x_{iD}) \tag{52}$$

and

$$V_i = \left(v_{i1}, v_{i2}, \ldots, v_{id,}, \ldots, v_{iD}\right). \tag{53}$$

The optimal position of the i^{th} particle , P_i, and the swarm , P_g, is given by [70]

$$P_i = \left(p_{i1}, p_{i2}, \ldots, p_{id,}, \ldots, p_{iD},\right) \tag{54}$$

and

$$P_g = \left(p_{g1}, p_{g2}, \ldots, p_{gd,} \ldots, p_{gD}\right). \tag{55}$$

If the objective function is to be minimised, the optimal position of the i^{th} particle is updated according to the relationship [70]

$$p_{i,t+1}^d = \begin{cases} x_{i,t+1}^d, if\ f\left(X_{i,t+1}\right) < f\left(P_{i,t}\right) \\ p_{i,t}^d, otherwise \end{cases}, \tag{56}$$

where $f()$ denotes the value of the predefined objective function at the given position, and $1 \leq d \leq D$, where D represents the dimensionality of the search space. The pre-defined objective function evaluates the performance of the particle in solving the optimisation problem [67].

The formula to update the velocity and position of each particle, based on the modified PSO algorithm by Shi and Eberhard [68], is given by [70]

$$v_{i,t+1}^d = [w \times v_{i,t}^d] + [c_1 \times rand \times (p_{i,t}^d - x_{i,t}^d)] + [c_2 \times rand \times (p_{g,t}^d - x_{i,t}^d)] \tag{57}$$

and

$$x_{i,t+1}^d = x_{i,t}^d + v_{i,t+1}^d, \tag{58}$$

respectively, where w represents the inertia weight [67], [68], [70]. The inertia weight can either be fixed or can be set to change dynamically over time [69]. Parameters c_1 and c_2 are both constants, known as the cognitive learning factor and the social learning factor, respectively. The *rand* represents a random number between 0 and 1 [67], [68], [70].

The velocity of the particles is updated iteratively so that the particles moves in a stochastic manner around the P_i and P_g during the search pattern [67]. The velocity of the particle in (57) can be split up into three terms, namely the inertia term, the cognitive term and the social term [68], [70]. The inertia term represents the influence that the particle's previous velocity has on its current velocity and models the resistance of the particle to change its velocity [68], [69], [70]. The inertia weight enables the local and global search ability to be controlled and balanced [68]. There is a trade-off associated with the local and global search [68]. The cognitive term incorporates the distance between the current position of the particle and the optimal position of the particle and considers the ability of the particle to think for itself and learn from its own experiences. The social term incorporates the distance between the current position of the particle and the optimal position of all the particles and considers the ability of the particle to collaborate with and learn from other particles [68], [70]. The cognitive and social term bring about the change in the velocity [68]. The constant c_1 and c_2 determine the contribution and the importance of the cognitive term and the social term in calculating the velocity of the particle [71]. Essentially, it determines how much each particle is pulled towards its personal best or the global best of the swarm respectively [69], [70], [71]. The procedure describing the iteration of the particles is illustrated in Fig. 9.

The formulation in (57) represents the global version of the PSO, that tracks the optimal position of each particle as well as the optimal position of the swarm. In the alternative local version of the PSO, the local version tracks the optimal position of each particle and the optimal position of the particles in its neighbourhood topology P_l. The global version and the local version of the PSO are therefore similar in the sense that $p_{g,t}^d$ is replaced with $p_{l,t}^d$ in the governing equations.

2.6.3 Advantages and disadvantages of particle swarm optimisation

The benefits of the PSO include its intuitive nature and ease in implementing the algorithm. Additionally, the PSO allows a wide range of functions to be optimised, i.e. the functions are not limited to continuous, differential or derivative functions. The PSO can also quickly converge to an optimal solution [70].

The PSO has several drawbacks. If a function has multiple local extremes, it is challenging to determine the optimal solution, since the particles are drawn to these local extremes, which may cause the PSO to converge prematurely. It is not certain that the PSO will find the global optima of the optimisation problem. Additionally, the PSO is not supported by an exhaustive theoretic foundation [70].

2.7 Clustering

2.7.1 Overview

Clustering is a procedure that attempts to group similar data points together, ensuring that data points which are placed in the same group are more alike than data points which are placed in other groups. Clustering is an unsupervised machine learning algorithm, i.e. the dataset used in clustering contains unlabelled data with regards to group information. The objective of clustering is to determine the hidden structure within an unlabelled dataset [72].

The clustering process can be summarised in the following steps [73], [74]:

- Select the entities that require clustering. It is important that the chosen dataset represents the cluster structure present in the population.
- Select the variables describing the entities that will be used in the clustering analysis.
- Decide if it is necessary to standardise the data or not. If the data is to be standardised, various standardisation approaches should be evaluated to find the favourable approach.
- Select a distance measure, defined as either a dissimilarity or similarity measure, to be used. These measures describe how close or separated objects are. The value of a similarity measure increases as two objects become closer in similarity, thus reflecting a direct relationship, while the value of a dissimilarity measure increases as two objects become further in similarity, reflecting an inverse relationship.
- Select a clustering algorithm, based on the characteristics and underlying structures of the dataset to be clustered.
- Determine the optimal number of clusters to be used. Several clustering algorithms require the number of clusters to be preselected.
- Interpret and evaluate the validity of the clusters obtained during analysis by applying field knowledge and determine whether the clustering results can be replicated.

2.7.2 Distance measures and distance matrices

As aforementioned, the aim of clustering is to ensure that observations which are placed in the same group are more similar than observations placed in other groups. Clustering analysis thus requires methods to calculate the pairwise distance between observations, called the distance measures or the (dis)similarity between a pair of observations. Numerous distance measures exist. The distance measure will influence the cluster configurations greatly, and the choice of a distance measure therefor represents a vital step in clustering [75]. It is important that the choice of distance measures reflects the similarities and differences of interest [72].

Some common distance measures include the Euclidean, Manhattan, Chebyshev and Minkowski distance measures.

The Euclidean distance measure calculates the square root of the sum of the squared differences between the features of two observations, x and y. It is defined by the relationship [76]

$$d_{euc}(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}\,, \tag{59}$$

where x and y are both vectors of length n, i.e. x and y are represented by n variables.

The Manhattan distance measure calculates the sum of the absolute differences between the features of two observations, x and y. It is defined by the relationship [76]

$$d_{man}(x,y) = \sum_{i=1}^{n}|(x_i - y_i)|\,. \tag{60}$$

The Chebyshev distance, also known as maximum value distance measure, calculates the maximum value of the absolute differences between the features of two observations, x and y. It is defined by the relationship [76]

$$d_{max}(x,y) = \max_i |x_i - y_i|\,. \tag{61}$$

The Minkowski distance is a generic distance measure given by the relationship [76]

$$d_{minkowski}(x,y) = \sum_{i=1}^{n}\left(|(x_i - y_i)|^{\frac{1}{p}}\right)^p\,, \tag{62}$$

where p denotes an integer. The Euclidean distance measure and the Manhattan distance measure are both specific cases of the Minkowski distance measure, where $p = 2$ and $p = 1$, respectively.

Correlation-based distance measures are also widely used. A correlation-based distance is defined as the difference between the constant 1 and the correlation coefficient. The Pearson correlation distance, the most used correlation-based distance measure, determines the degree of linear relationship between elements x and y. It is defined by the relationship [74]

$$d_{cor}(x,y) = 1 - \frac{\sum_{i=1}^{n}(x_i - \bar{x})\,(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}\,, \tag{63}$$

where $\bar{x}$ and $\bar{y}$ represent the mean of x and y, respectively.

The distance matrix is a $n \times n$ matrix, where n denotes the number of observations, containing the distances between all the pairs of observations. The matrix element at row i and column j will thus be the distance between observation i and observation j of the dataset. There are zeroes on the diagonal, because it refers to the distance between an observation and itself. The matrix is symmetric, because the distance between two observations stays constant regardless of the order in which it is referred to [72].

2.7.3 Standardisation of data

The value of the distance measures is dependent on the measurement scales of the variables of the observations [75]. If each observation in the dataset comprises of n variables or features, it is represented as a coordinate in a n-dimensional dataspace. Distances are then calculated between these coordinates. If the range of one variable is much larger than the range of another variable, the larger variable will have a greater influence on the distance measure, which may lead to incorrect clustering results [75], [77].

Scaling the data is recommended when the variables representing the observations are measured in different scales [75]. Scaling converts the original measurements to unitless variables. The aim of scaling the data is to avoid dependence on measurement units when calculating the distance measure and to make the variables comparable [75], [77]. There are two scaling approaches that are widely used, namely normal standardisation and min-max normalisation.

Normal standardisation transforms the variables so that it has as mean of 0 and a standard deviation of 1. It is defined by the relationship [75], [77], [78]

$$x(s) = \frac{x(i) - mean(x)}{sd(x)}, \tag{64}$$

where x denotes one of the variables representing an observation, $x(i)$ denotes the i^{th} value of the variable x and $x(s)$ denotes the scaled representation of $x(i)$. Furthermore, $sd(x)$ and $mean(x)$ denote the standard deviation and the mean of variable x, respectively.

Min-max normalisation standardises the entire scale of all the variables by transforming $x(i)$ to fall between $[0,1]$. It is defined by the relationship [78]

$$x(s) = \frac{x(i) - min(x)}{max(x) - min(x)}, \tag{65}$$

where $min(x)$ and $max(x)$ denotes the minimum and maximum of variable x, respectively.

2.7.4 Partitioning clustering algorithms

2.7.4.1 Overview

Partitioning clustering methods are characterised by the formation of distinct and non-overlapping clusters. Partitioning clustering methods require initial starting partitions to be selected. These starting partitions are either generated randomly or specified by the user. These methods also require that the number of clusters is chosen in advance, thus forcing outliers to become part of one of the final clusters [73], [74]. The objective of these methods is to optimise a certain criterion function via its cluster assignments, through an iterative process [79]. Some commonly used partitioning clustering methods include k-means clustering and Partitioning Around Medoids (PAM).

2.7.4.2 K-means clustering algorithm

The objective of the k-means algorithm is to assign the n observations of a dataset to a fixed number of clusters, k. Thus, it is required that the number of clusters should be preselected by the user. Each of the k clusters are represented by the cluster centroid, which can be either a real or imaginary location. The cluster centroid is defined as the mean of all the data points within a cluster [80]. Observations are iteratively assigned to the closest cluster based on the similarity between the data point and the cluster centroids [74]. The procedure followed during k-means clustering is as follows [72], [75]:

- Select the number of clusters, k, to be generated.
- Randomly generate initial estimates for the k cluster centroids or select it via user specification.
- Allocate each observation to a cluster based on its nearest cluster centroid according to the chosen distance metric.
- Calculate the new cluster centroids as the average of all the observations belonging to the cluster.
- Repeat the two previous steps until a stopping criterion is met. The stopping criterion entails that either no change in cluster assignments are observed or that the maximum number are iteration are reached.

The initial k cluster centroids are generally randomly generated. The result of the k-means algorithm is, however, sensitive to the initial choices of cluster centroids. It is therefore vital to run the k-means algorithm using multiple, randomly generated, initial cluster centroids and select the initial centroids that produces the best results. If the number of variables for each observation increases, the space representing the observations becomes more complex and thus more randomly generated initial cluster centroids are required [72].

The k-means algorithm seeks to define k clusters with the aim of minimizing the total within-cluster sum of squares, which is defined as [81]

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 , \tag{66}$$

where k represents the number of clusters and j represents thej^{th} cluster. The parameter n denotes the number of observations within the cluster j and $\mathrm{x}_\mathrm{i}^{(\mathrm{j})}$ denotes the i^{th} observation of cluster j. Additionally, $\left\| \mathrm{x}_\mathrm{i}^{(\mathrm{j})} - \mathrm{c}_\mathrm{j} \right\|^2$ denotes the chosen distance measure used to calculate the distance between $\mathrm{x}_\mathrm{i}^{(\mathrm{j})}$ and its corresponding cluster centroid c_j.

The standard algorithm used for k-means clustering is the Hartigan-Wong algorithm. For the Hartigan-Wong algorithm, the total within-cluster sum of squares is defined as the sum of the squared Euclidean distance between the observations and the respective cluster centroid, i.e. [75]:

$$W_k = \sum_{k=1}^{C_k} \sum_{i:x_i \in C_k} (x_i - \overline{x_{C_k}})^2 , \tag{67}$$

where k represents the number of clusters, x_i is the i^{th} observation assigned to cluster C_k and $\overline{x_{C_k}}$ is the cluster centroid of cluster C_k.

2.7.4.3 Partitioning around medoids

The Partitioning Around Medoids (PAM) algorithm, which is similar to the k-means algorithm, also iteratively groups n observation within a dataset into k clusters. The number of clusters, therefore, has to be preselected by the analyst. For PAM clustering, each cluster is represented by a cluster medoid, which is one of the data points within the cluster, instead of a cluster centroid as for k-means. The medoid is the data point within the cluster with the smallest average dissimilarity between it and all other data points. The medoid corresponds to the most central point in the cluster [75].

The aim of the PAM algorithm is to determine these k medoids such that the total dissimilarity of all the dataset elements to their nearest medoid is minimised. This can be done by minimizing the objective function given by [82]

$$\sum_{i=1}^{n} \min_{t=1,\dots,k} d(x_i, m_t) , \qquad (68)$$

where k represents the number of clusters, m_t represents the t^{th} medoid, n is the number of elements in the dataset and $\{x_i, i = 1, \dots, n\}$ is the dataset elements [82]. The dissimilarity between the observation x_i and its nearest medoid m_t is represented by $d(x_i, m_t)$.

The procedure followed during PAM clustering is as follows [75]:

- Select the number of clusters, k, to be generated.
- Randomly generate initial estimates for the k cluster medoids or select it via user specification.
- Allocate each observation to a cluster based on its nearest cluster medoid according to the chosen distance metric.
- Determine the new cluster medoids for each cluster as the observation within a cluster which has the smallest average dissimilarity between it and all the other data points within the cluster.
- Repeat the two previous steps until the medoids representing each cluster does not change.

The PAM algorithm is more robust than the k-means algorithm because it uses medoids instead of centroids to represent clusters. Thus, the PAM algorithm is affected less by outliers and noise [82].

2.7.5 Hierarchical clustering algorithms

2.7.5.1 Overview

Hierarchical clustering methods generate clusters via a recursive procedure. During the clustering process, smaller clusters are recursively merged to form larger clusters or larger clusters are recursively divided into smaller clusters [79]. Hierarchical clustering does not require the number of clusters to be preselected [75]. Hierarchical clustering methods consist of two types, namely agglomerative and divisive clustering. The results of hierarchical clustering are generally represented by a dendrogram. A dendrogram is a tree-like structure that illustrates the order in which the clusters were merged or divided during each step [75].

2.7.5.2 Agglomerative clustering

The agglomerative clustering method starts with each data point considered as its own cluster and, thereafter, similar clusters are iteratively merged until only one cluster which comprises the entire dataset is formed [73]. The procedure followed during agglomerative clustering is as follows [83]:

- Consider each data point as a single cluster.
- Calculate the distance matrix for the current clusters.
- Merge the two clusters with the smallest distance between them.
- Repeat the two previous steps until only one cluster comprising all the data points remain.

As clusters can contain one or more observations, it is important to determine how the distance between single observations will be defined, as well as how the distance between clusters with two or more observations will be defined [72].

2.7.5.2.1 Distance (dissimilarity) between clusters with single observations

The distance between clusters comprising of a single observation each is calculated using distance measures, as defined in section. 2.7.2.

2.7.5.2.2 Distances between clusters with two or more observations

There are various methods to calculate the distance between clusters with two or more observations, known as non-singleton clusters. These methods are called linkage criteria. Some common types of linkage criteria include the following [72]:

- *Complete (or maximum) linkage:* For complete linkage all pairwise distances between observations in cluster A and observations in cluster B is calculated. The complete linkage then uses the maximum value of all the pairwise distances as the distance between cluster A and cluster B.
- *Single (or minimum) linkage:* For single linkage all pairwise distances between observations in cluster A and observations in cluster B is calculated. The single linkage then uses the minimum value of all the pairwise distances as the distance between cluster A and cluster B.
- *Average linkage:* For average linkage all pairwise distances between observations in cluster A and observations in cluster B is calculated. The average linkage then uses the average of all the pairwise distances as the distance between cluster A and cluster B.
- *Centroid linkage:* For centroid linkage, the distance between the centroid of cluster A and the centroid of cluster B is used as the distance between cluster A and cluster B.
- *Ward's method:* For the Ward's method, the distance between cluster A and cluster B is defined as the sum of squares for the cluster obtained from combining cluster A and cluster B minus the sum of squares for cluster A and cluster B, separately.

The formula for each linkage criterion is included in Table 5, where $\boldsymbol{L(A,B)}$ denotes the linkage value between cluster $\boldsymbol{A}$ and cluster $\boldsymbol{B}$ and $\boldsymbol{d(x_a, x_b)}$ denotes the distance between observation $\boldsymbol{x_a}$ and $\boldsymbol{x_b}$. The advantages and disadvantages of each linkage criterion is shown in Table 6 [72].

Table 5: Linkage criteria for agglomerative clustering [72].

Linkage criterion	Formula
complete linkage	$L(A,B) = \max\{d(\boldsymbol{x}_a, \boldsymbol{x}_b) : \boldsymbol{x}_a \in A,\ \boldsymbol{x}_b \in B\}$
single linkage	$L(A,B) = \min\{d(\boldsymbol{x}_a, \boldsymbol{x}_b) : \boldsymbol{x}_a \in A,\ \boldsymbol{x}_b \in B\}$
average linkage	$L(A,B) = \frac{1}{\text{no. of pairs}} \sum_{\{a,b\}:\boldsymbol{x}_a \in A,\, \boldsymbol{x}_b \in B} d(\boldsymbol{x}_a, \boldsymbol{x}_b) = \frac{1}{\lvert A\rvert \lvert B\rvert} \sum_{\{a,b\}:\boldsymbol{x}_a \in A,\, \boldsymbol{x}_b \in B} d(\boldsymbol{x}_a, \boldsymbol{x}_b)$
centroid linkage	$L(A,B) = d(\overline{\boldsymbol{x}}_A, \overline{\boldsymbol{x}}_B)$ where $\overline{\boldsymbol{x}}_A = \frac{1}{\lvert A\rvert} \sum_{a:\boldsymbol{x}_a \in A} \boldsymbol{x}_a$
Ward's linkage	$L(A,B) = SS(A,B) - (SS(A) + SS(B))$ where $SS(A) = \sum_{a:\boldsymbol{x}_a \in A} (\boldsymbol{x}_a - \overline{\boldsymbol{x}}_A)^2$

Table 6: Advantages and disadvantages of linkage criterion [72].

Linkage criterion	Advantages	Disadvantages
complete linkage		sensitive to outliers
single linkage	can find irregular-shaped clusters	sensitive to outliers
average linkage	generates clusters with small within-cluster varation	
centroid linkage	robust to outliers	
Ward's linkage		

2.7.5.3 Divisive clustering

Divisive clustering starts with all data points in one large cluster, as opposed to agglomerative clustering, and, thereafter, clusters that are not similar are iteratively separated until each cluster is considered as a single cluster. The idea behind agglomerative clustering can easily be adapted for divisive clustering [83].

2.7.6 Fuzzy c-means clustering

For c-means clustering each observation in the dataset can belong to more than one cluster. The observations in the dataset is given by the set [84]

$$X = \{x_i, i = 1, \dots, N\} \tag{69}$$

and the clusters is represented by the set [84]

$$C = \{C_j, j = 1, \dots, k\}. \tag{70}$$

A dataset element x_i belongs to cluster C_j with a membership degree u_{ij}. The membership degree u_{ij} ranges from 0 to 1. If N is the number of elements in the dataset and K is the number of clusters, the membership degrees of all the elements to all the clusters is contained in a $N \times K$ matrix represented by [84]

$$U = \{u_{ij}\}_{i,j}^{NK}, \tag{71}$$

where the element in the i^{th} row and j^{th} column is u_{ij}.

The aim of the fuzzy c-means algorithm is to minimise the objective function given by [84]

$$J_m(U, R) = \sum_{i=1}^{N} \sum_{j=1}^{K} {u_{ij}}^m \left|x_i - C_j\right|^2, \ 1 \leq m < \infty, \tag{72}$$

where m represents the fuzzifier parameter which determines the fuzziness of the clusters, with $m \in \mathbb{R}$, and $\left|x_i - C_j\right|$ represents the Euclidean distance between the i^{th} dataset element, x_i, and the cluster centroid of the j^{th} cluster, C_j. The fuzzy c-means algorithm behaves like the k-means at the limit $m \to 1$ [84].

The objective function J_m has to satisfy the conditions given by

$$u_{ij} \in [0,1]; \sum_{j=1}^{k} u_{ij} = 1, \forall i; 0 < \sum_{i=1}^{N} u_{ij} < N, \ \forall N. \tag{73}$$

The procedure followed during c-means clustering is as follows [84]:

1. Randomly initialise the membership degree matrix U^0.

2. Calculate the cluster centroids using the relationship [84]

$$C_j = \frac{\sum_{i=1}^{N} {u_{ij}}^m x_i}{\sum_{i=1}^{N} {u_{ij}}^m} . \tag{74}$$

3. Calculate the new membership degree matrix by updating the values of the membership degrees according to the relationship

$$u_{ij} = \frac{1}{\sum_{k=1}^{K} \left(\frac{x_i - c_j}{x_i - c_k}\right)^{\frac{2}{m-1}}} . \tag{75}$$

4. Repeat steps 2 and 3 until

$$\left| U^{k+1} - U^k \right| < e . \tag{76}$$

The c-means algorithm improves on cases present in crisp clustering, i.e. having disjoint clusters, where an element that is equally distanced from cluster centroids is randomly assigned to one of these clusters [84].

2.7.7 Determine optimal number of clusters

2.7.7.1 Overview

As mentioned previously, partitioning clustering algorithms requires the number of clusters to be preselected by the analyst [74]. The optimal number of clusters is subjective, and it depends on the selected partitioning clustering method as well as the input parameters of the method, such as the distance measure.

A method is, therefore, required to determine the optimal number of clusters in a simple and algorithmic fashion [74]. There are two direct methods that can be used to determine the optimal number of clusters, namely the elbow method and the average silhouette method. Direct methods entail determining the number of clusters by optimizing a certain criterion [75]. Another method involves studying the dendrograms resulting from the hierarchical clustering algorithms to see if it reveals an optimal number of clusters [75].

2.7.7.2 Elbow method

The elbow method calculates and plots the total Within-Cluster Sum of Squares (WSS), as defined in (66), as a function of the number of clusters k. The WSS measures how compact or separated the objects are in a cluster. A small value of the WSS indicates compact clusters which is desirable. The optimal number of clusters is the k which corresponds to a bend in the curve. The bend signifies the point where a further increase in k will not significantly change the WSS [75].

2.7.7.3 Average silhouette method

The silhouette method calculates and plots the average silhouette value, which will be defined in section 2.7.8.2, as a function of the number of clusters k. Briefly, the average silhouette value measures whether objects were assigned to the correct cluster or not. The optimal number of clusters is the k value which maximises the average silhouette value [75].

2.7.8 Cluster validation

2.7.8.1 Overview

It is vital to evaluate the clustering results obtained for a dataset to determine whether it is acceptable or not. The clustering results obtained for a dataset by varying the combinations of the clustering methods, the distance measures and the linkage criteria, where applicable, shows a great deal of variety. Additionally, clustering is an unsupervised machine learning technique.

There are various validation measures which can be used to determine the validity of the clustering results, as well as to compare the different clustering results obtained by varying the combinations of the clustering methods, the distance measures and the linkage criteria, where applicable, in order to find the optimal clustering methodology for the given dataset. The validation measures are only applied to crisp clustering. Three validation measures, namely the silhouette coefficient, the Dunn index and the Calinsky-Harabasz, are presented [79].

2.7.8.2 Silhouette plot and silhouette coefficient

A silhouette plot, i.e. a plot containing the silhouette value of each observation in the dataset element, is used to evaluate the results of the clustering analysis. The silhouette value of an observation provides information about the similarity of the observation to the cluster it is assigned to, i.e. cohesion, compared to the similarity of the observation to the neighbouring clusters, i.e. separation. Specifically, the silhouette value of an observation is given by the relationship [72]

$$s_i = \frac{(b_i - a_i)}{max\{a_i, b_i)}, \tag{77}$$

where a_i and b_i represent the mean intra-cluster distance and the mean nearest-cluster distance of the i^{th} dataset observation, respectively [74]. The mean intra-cluster distance is defined as the average of all the pairwise distances between the i^{th} observation and all the other observations assigned to the same cluster. The mean nearest cluster distance is defined as the minimum value of the average distances between the i^{th} observation and the observations in the neighbouring clusters, $B(i,k)$. The mean nearest cluster distance is given by the relationship [72], [74]

$$b_i = min_k\{B(i,k)\}. \tag{78}$$

The silhouette value lies between $[-1, 1]$. If $a_i \ll b_i$, the value of s_i is close to 1 indicating that the observation is correctly assigned to its cluster, because a small a_i indicates that the i^{th} observation is similar to the other observations in the same cluster and a large b_{i} indicates that the i^{th} observation is dissimilar to observations contained in other clusters. If $b_i \ll a_i$, the value of s_i is close to -1. According to the same logic, a value close to -1 indicates that the observation is incorrectly assigned. If many observations, therefore, have a silhouette value close to -1. it indicates that either the wrong clustering method might have been used or the number of clusters are incorrect. Additionally, if $a_i \approx b_i$, s_i is close to 0 which indicates that the observation is on the boundary of two adjacent clusters [72].

The average silhouette value and the silhouette plots can be used to determine the optimal number of clusters for a given dataset. The average silhouette value of all the observations in a cluster indicates how dense the observations are in the cluster. Thus, the average silhouette value over all observations in the dataset indicates whether the cluster configuration is correct or not. If the silhouette plot contains clusters

which has much narrower silhouettes when compared to other clusters, it indicates that a poor choice regarding the number of clusters was made [72].

2.7.8.3 Dunn index

The Dunn index is given by the relationship [79]

$$D = \min_{i=1,\dots,k} \{ \min_{j=1,\dots,k} (\frac{d(c_i,c_j)}{\max_{m=1,\dots,k} diam(c_m)}) \}, \quad (79)$$

where k represents the number of clusters, and c_i and c_j represents the i^{th} and j^{th} cluster, respectively. Additionally, $d(c_i, c_j)$ represents the inter-cluster distance between cluster c_i and cluster c_j and $diam(c_i)$ represents the intra-cluster diameter of cluster c_i. The inter-cluster distance between cluster c_i and c_j is given by the relationship [79]

$$d(c_i, c_j) = \min_{x \in c_i, y \in c_j} \{d(x,y)\}, \quad (80)$$

where $d(x,y)$ represents the distance between two dataset elements, x and y, that are in different clusters. The intra-cluster diameter of cluster c_j is defined by the relationship [79]

$$diam(c_i) = \max_{x \in c_i, y \in c_j} \{d(x,y)\}, \quad (81)$$

where $d(x,y)$ represents the distance between two elements, x and y, that are in the same cluster.

The Dunn index is proportional to inter-cluster distances and inversely proportional to intra-cluster diameters. Clusters that are well separated are characterised by large distances between the distinct clusters, i.e. a large inter-cluster distance, and small intra-cluster diameters. Better cluster configurations are, therefore, represented by a large Dunn index.

The Dunn index has a few drawbacks. The Dunn index has a relatively long computation time. Additionally, as the intra-cluster diameter is used in the calculation, the Dunn index is very sensitive to noise, because a noisy environment can increase the intra-cluster diameter [79].

2.7.8.4 Caliński-Harabasz index

The Caliński-Harabasz (CH) index makes use of the average between-cluster sum of squares and the average within-cluster sum of squares to evaluate the validity of clusters. The Caliński-Harabasz index is given by the relationship [85]

$$CH(k) = \frac{(\frac{trace\ B}{K-1})}{(\frac{trace\ W}{N-K})} for\ K \in \mathbb{N}, \quad (82)$$

where N represent the number of elements in the dataset and K represents the number of clusters [86]. Additionally, $\mathrm{trace\ B}$ and $\mathrm{trace\ W}$ are the average between-cluster sum of squares and the average within-cluster sum of squares, respectively. The average between-cluster sum of squares is defined as the sum of the squares of the differences between the centroid of each cluster and the centroid of the dataset, weighted by the corresponding size of the cluster [85], [87]. The average within-cluster sum of squares is defined as the squared differences between all the observations within a cluster and their respective cluster centroids [87]. The average between-cluster sum of squares is given by the relationship [85]

$$trace\ B = \sum_{k=1}^{K} |C_k|\, \|\overline{C_k} - \bar{x}\|^2 , \qquad (83)$$

where C_k and $\overline{C_k}$ represents the number of points and the cluster centroid of the k^{th} cluster respectively, and $\bar{x}$ is the centroid of the entire dataset. The average within-cluster sum of squares is given by the relationship [85], [86]

$$trace\ W = \sum_{k=1}^{K} \sum_{i=1}^{N} \|x_i - \overline{C_k}\|^2 , \qquad (84)$$

where x_i denotes the i^{th} element in the dataset [86].

A minimised Caliński-Harabasz index represents optimal cluster configurations [74].

3 Ramp Detection Models

3.1 Overview

This chapter provides the implementation, results and comparison of the various ramp detection models investigated in the project. This chapter is divided into 3 parts:

- *Section 3.2:* The details of the measured wind power data used for the study, as well as the necessary processing performed on the data, is provided.
- *Section 3.3-3.8:* The implementation of the different ramp detection methods investigated in this study are presented, and the corresponding results is provided and discussed to gain insights into the ramp detection process of each method. Table 7 summarises the different ramp detection models implemented and discussed in this chapter, which can be classified into two groups, namely the classical ramp detection models and the newly proposed ramp detection models. A more detailed discussion of the results of the Swinging Door Algorithm (SDA), L1-ramp detect with Sliding Window (L1-SW), Optimised Swinging Door Algorithm (OpSDA) and the multi-parameter segmentation algorithm are provided. The implementation and results of the remaining two ramp detection models, i.e. a multi-parameter segmentation algorithm with particle swarm optimisation and the regression-based segmentation algorithms, are included for exploratory purposes only. Accordingly, no detailed evaluation or comparison of the results is included.
- *Section 3.9:* The results for the comparison of the sets of ramps detected by the SDA, L1-SW, OpSDA and the multi-parameter segmentation algorithm for optimal parameter values is presented.

Table 7: Summary of the different ramp detection algorithms investigated in this study.

Classical ramp detection models	**Newly proposed ramp detection models**
Swinging door algorithm	A multi-parameter segmentation algorithm
L1-ramp detect with sliding window	A multi-parameter segmentation algorithm with particle swarm optimisation
Optimised swinging door algorithm	Regression-based segmentation algorithms

In this report, a significant ramp is defined as a change in wind power output that is greater than or equal to 20% of the maximum wind power output without restricting the ramp duration. The magnitude threshold is consistent with values found in other work. It is, however, easy to apply other ramp definitions for ramp detection to produce different results.

All the ramp detection methods are implemented in R, except for the L1-SW and the OpSDA, which are implemented in MATLAB.

3.2 Data description

The measured wind power data used for the study is from a utility size wind farm located near Koekenaap, Western Cape, South-Africa. The wind farm falls within the Matzikama municipality. The data set contains 157680 samples representing actual wind power production from 01 January 2017 at 00:00 to 31 December 2019 at 23:50, and the data has a temporal resolution of 10 minutes. The windfarm has a nominal capacity of 100 MW. The respective ramp detection models are applied to the measured wind power data to extract the corresponding significant ramps. The extracted ramping information is used to evaluate and compare the different methods, and the data is also used as input for the statistical analysis and clustering methodology. Although the ramp detection models are applied to the entire data set, the results of the segments and significant ramps identified via the respective ramp detection models are only

shown for the first 400 discrete samples, to clearly illustrate the effect of varying the parameters of each ramp detection model. It is important to note that the results of the statistical analysis and clustering methodology are, thus, for the entire data set, i.e. not just for the 400 samples used to visually present the detected ramps.

Pre-processing of the data to account for noise is not required, since the permissible noise threshold is already accounted for by each ramp detection model. Specifically, the tunable parameter, epsilon, in the SDA and OpSDA defines the threshold for ramp variations, and the tunable parameter, γ, in the multi-parameter segmentation algorithm characterises the permissible vertical fluctuation. The λ parameter in the linear trend fitting technique is used to remove insignificant noise and emphasise appropriate trends in the data. The pre-processing of the data, therefore, only includes changing negative power values to zero since negative power values are not allowed, and missing power values are addressed by substituting it with zeroes. The data set is also normalized with respect to the maximum measured power contained in the data set before the application of the respective ramp detection models.

3.3 Swinging door algorithm

3.3.1 Implementation

The Swinging Door Algorithm (SDA) extracts ramps from a signal in a piecewise linear fashion [16]. The ramps are regarded as segments in which the wind power is considered to follow a single trend [53]. The aim when extracting ramps from measured data is, therefore, to disregard insignificant fluctuations or measurement noise in the data. The aim of extracting ramps from simulated data is to disregard insignificant fluctuations. In view of the above, significant linear ramps are extracted from the signal [16].

The SDA incorporates one tunable parameter, ε, which defines the width of the swinging door. Essentially, this parameter specifies the threshold for the ramp trend, i.e. the sensitivity to (or conversely tolerance for) ramp variations which include noise or insignificant fluctuations. If the tunable parameter is small, the tolerance for ramp variations is also small, and the SDA will extract many small ramps, because the threshold is easily violated. However, if the tunable parameter is large, the tolerance for ramp variations is also large. The SDA will, thus, extract a few large ramps as the threshold is not easily violated. A large variation is required to violate the threshold.

Fig. 10 illustrates the implementation of the SDA based on swinging doors, and a brief summary of the implementation follows. For a new iteration of the SDA, threshold doors of width epsilon are placed above and below the starting point of the new ramp respectively, and the points at the tail end of the threshold doors become the respective door hinges. During the iteration of the algorithm, a new point is iteratively acquired and the current positions of the doors are established accordingly. —To elaborate, at each point conditions are checked to determine if the swinging doors open to include that point or if the swinging doors remain in their respective positions. Regarding the conditions, the top door can only swing open in the anti-clockwise direction to include a point and the bottom door can only swing open in the clockwise direction to include a point. — Furthermore, the positions of the swinging doors determine whether the iteration of the algorithm continues or stops. The current iteration of the algorithm stops, i.e. the ramp is found, when the angle between the swinging doors are greater than or equal to parallel, as it means the threshold was exceeded. Accordingly, a new iteration of the SDA will start at the end of the previous ramp [16].

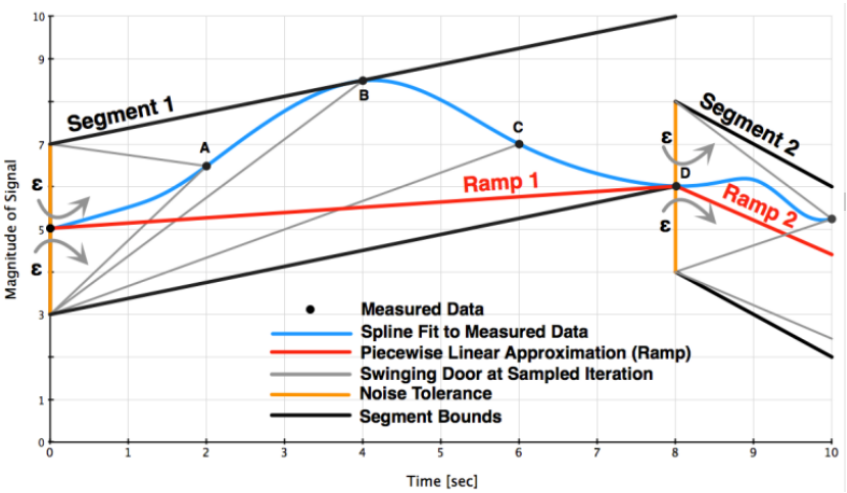


Fig. 10: The implementation of the swinging door algorithm for extracting ramps in a power signal [16].

The implementation of the SDA according to Fig. 10 follows. The time series point on the y-axis is the starting point of the new ramp. A new point, point A, is obtained and the conditions are examined to determine if the top door or bottom door opens to include point A. Both conditions are met, and the door positions are updated accordingly. Lines are respectively drawn from the top and bottom door hinges to point A to indicate the current positions of the doors. The lines intersect at point A. Next, point B is obtained and in base of the same procedure as for A, lines are drawn from the respective door hinges to point B and intersect at point B. Once more, point C is obtained and the conditions to check whether the top door or bottom door open is tested. An inflection point occurred, therefore, only the bottom door opens to include point C while the top door remains in its existing position as per the conditions. A line is drawn from the bottom door hinge to point C and the line from the top door hinge to point B is extended above point C. The extension of both these lines will result in an intersection at some later point and so the iteration continues. Point D is obtained and again the conditions are tested to determine whether the top door and bottom door can open. Similarly, only the bottom door position is updated while the top door position remains the same and the modified lines are drawn. Any further extension of these lines, however, does not intersect at any point in the future, i.e. the angle between the swinging doors are greater than or equal to parallel. Thus, the threshold has been exceeded and the current ramp iteration terminates at D. Although the threshold was violated somewhere between point C and point D, point D becomes the end point of the ramp because of the discrete nature of the signal. A new iteration of the SDA will also start at D [16].

After the SDA is applied to the wind power time series to extract the linear ramps, a user specified definition of a significant ramp is used to identify the wind power ramp events present in the signal [16].

The advantages of using the SDA include the structural and computational simplicity of the algorithm as well as the ability to produce robust results in the presence of noise. Conversely, the limitation of this method is that there is no clear definition of what an insignificant fluctuation is when considering actual/measured data. According to Florita *et al.* [16], two applications can be used to define an insignificant fluctuation. Firstly, the definition of an insignificant fluctuation is dependent on the accuracy of the measurement device, as characterised by the distribution of the measurement uncertainty.

Secondly, it is dependent on the usage and importance of the measure, which is characterised by the economics of the power systems and its relative significance in driving operations.

3.3.2 Ramp detection results

The ramp detection results of the SDA are only shown for the first 400 discrete samples to illustrate the effect of varying epsilon. Fig. 11 to Fig. 15 depict the segments and significant ramps extracted via the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% and the selected ramp definition. The SDA extracts significant linear ramps from the signal, i.e. segments in which the wind power is considered to follow a single trend.

It is evident that as the value of epsilon decreases, i.e. the threshold for ramp variations decreases, the magnitude and duration of the segments used to linearly approximate the signal also decreases, since the threshold is more easily violated. This in turn increases the number of segments and ensures a more accurate approximation of the signal. Epsilon, therefore, introduces a trade-off between approximation accuracy and the number of segments, which is expected.

The detected ramps correspond to the segments extracted by the SDA that satisfy the selected ramp definition. Accordingly, epsilon affects the magnitude and duration of the detected ramps and in turn also the number of detected ramps, i.e. increasing the value of epsilon results in the detection of more ramps since larger segments are extracted via the SDA. It is recognised that the SDA does not successfully extract the start- and end-points of the segments. Specifically, the SDA fails to capture the peaks of the signal. This mostly happens when the start of a new ramp is found before an inflection point, and the inflection point is incorrectly considered to be part of a variation instead of representing a ramp end due to the defined threshold for ramp variations.

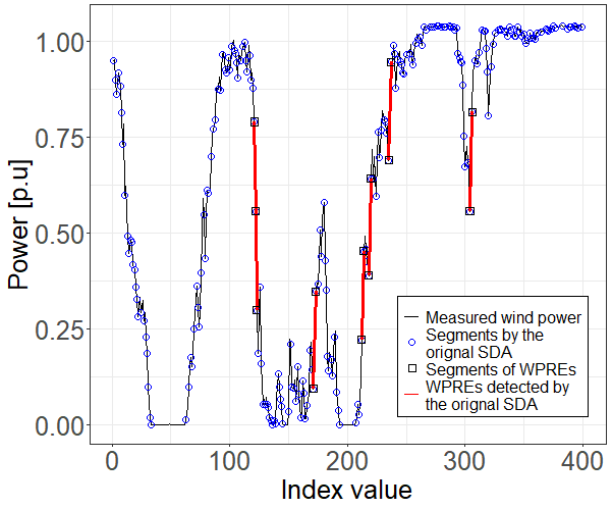


Fig. 11: Results of the ramp detection via the SDA for $\varepsilon = 0.001$ and $P_{val} = 0.2$.

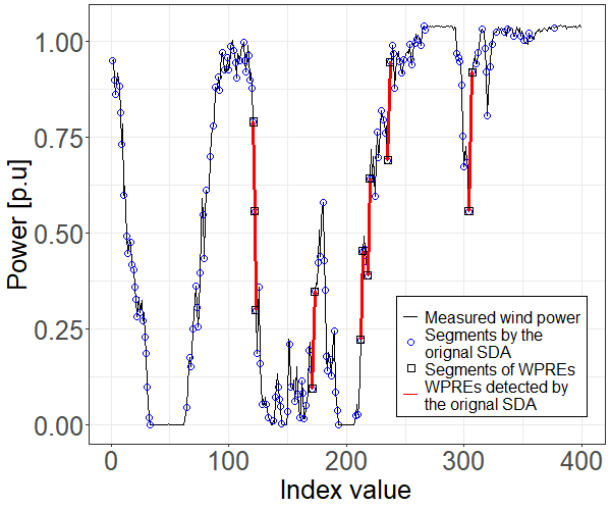


Fig. 12: Results of the ramp detection via the SDA for $\varepsilon = 0.009$ and $P_{val} = 0.2$.

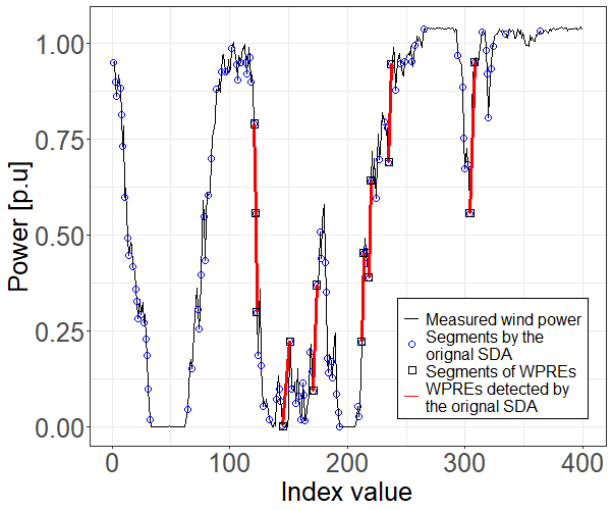


Fig. 13: Results of the ramp detection via the SDA for $\varepsilon = 0.02$ and $P_{val} = 0.2$.

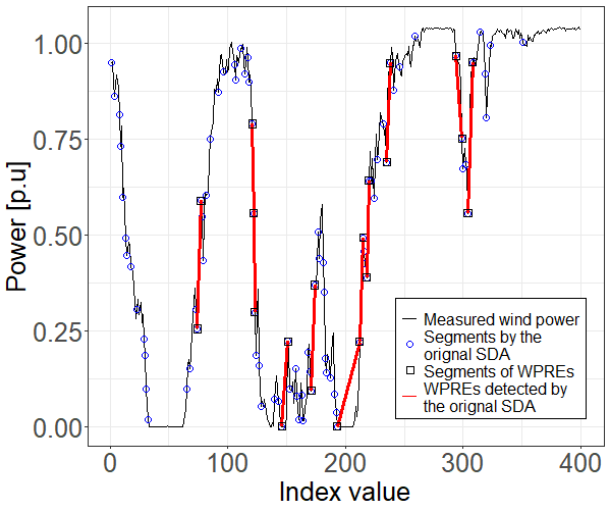


Fig. 14: Results of the ramp detection via the SDA for $\varepsilon = 0.03$ and $P_{val} = 0.2$.

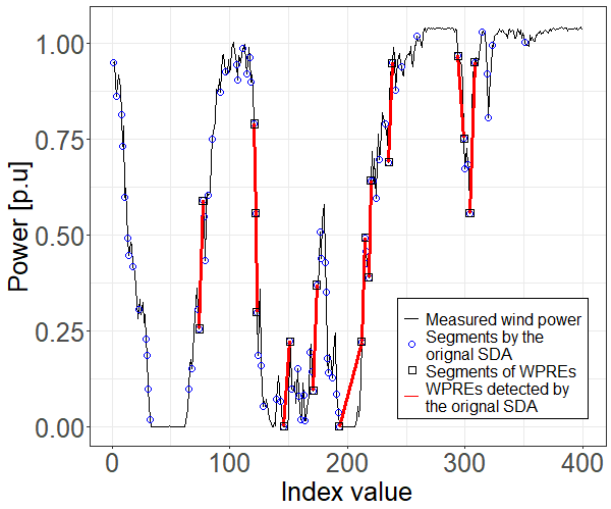


Fig. 15: Results of the ramp detection via the SDA for $\varepsilon = 0.05$ and $P_{val} = 0.2$.

3.3.3 Statistical analysis of key ramp features

3.3.3.1 Ramp duration

Fig. 16 and Fig. 17 depict the Probability Density Functions (PDFs) of the ramp duration of the upward and downward ramps, respectively, detected using the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%. It is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the SDA for the given parameter values will have a short duration rather than a long duration.

The probability density functions for the upward and downward ramps differ. The upward ramps detected by the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% have a mean ramp duration of 22.84 min, 28.72 min, 39.01 min, 49.58 min and 77.07 min, respectively. The maximum duration observed for the upward ramps

detected using the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% is 510 min, 740 min, 770 min, 770 min and 990 min, respectively. The minimum duration observed for the upward ramps detected using the SDA for the various values of epsilon is 20 minutes, respectively. Furthermore, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, 75% of the upward ramps have a duration less than or equal to 20 min, 30 min, 50 min, 60 min and 90 min. Conversely, the downward ramps detected by the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% have a mean ramp duration of 11.09 min, 12.34 min, 16.41 min, 30.90 min and 68.38 min, respectively. Additionally, the maximum duration observed for the downward ramps detected using the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% is 70 min, 160 min, 260 min, 370 min and 850 min, respectively. The minimum duration observed for the downward ramps detected using the SDA for the various values of epsilon is 10 minutes, respectively. Furthermore, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, 75% of the downward ramps have a duration less than or equal to 10 min, 10 min, 10 min, 40 min and 90 min.

Therefore, based on Fig. 16 and Fig. 17 as well as the information mentioned above, it is evident that the downward ramps detected by the SDA generally have a shorter duration than the upward ramps detected by the SDA. Additionally, if the size of epsilon decreases, the duration of the detected upward and downward ramps also decreases. This is consistent with the trade-off introduced by epsilon.

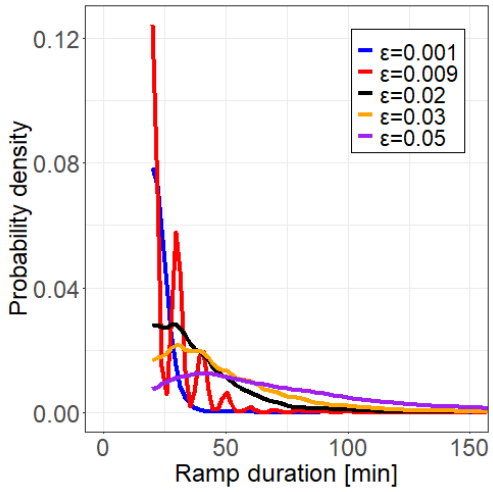


Fig. 16: Probability density functions of the ramp duration of upward ramps detected by the SDA for various parameter values.

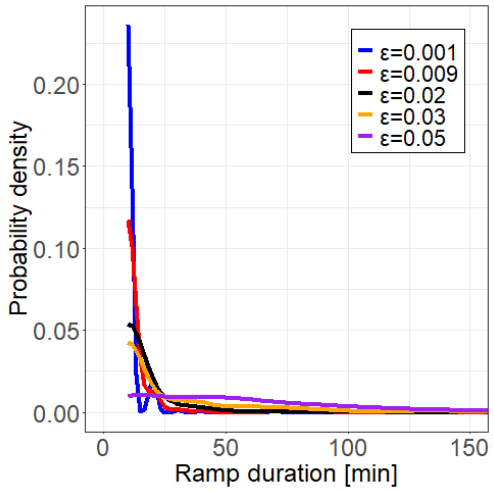


Fig. 17: Probability density functions of the ramp duration of downward ramps detected by the SDA for various parameter values.

3.3.3.2 Ramp magnitude

Fig. 18 and Fig. 19 depict the Probability Density Functions (PDFs) of the ramp magnitude of the upward and downward ramps, respectively, detected using the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%. It is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the SDA will have a small magnitude rather than a large magnitude.

The PDFs for the upward ramps differ from the PDFs for the downward ramps. The upward ramps detected by the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% have a mean ramp magnitude of 0.2984 p.u., 0.2984 p.u., 0.3023 p.u., 0.3103 p.u. and 0.3314 p.u., respectively. The maximum magnitude observed for the upward ramps detected using the SDA for the various values of epsilon is 1.0041 p.u. The minimum magnitude observed for the upward ramps detected using the SDA for the various values of epsilon is 0.20

p.u. Furthermore, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, 75% of the upward ramps have a magnitude less than or equal to 0.3174 p.u., 0.3214 p.u., 0.3341 p.u., 0.3484 p.u. and 0.3818 p.u. Conversely, the downward ramps detected by the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% have a mean ramp magnitude of 0.3020 p.u., 0.3028 p.u., 0.2987 p.u., 0.2845 p.u. and 0.2735 p.u., respectively. The maximum magnitude observed for the downward ramps detected using the SDA for the various values of epsilon is 1.0041 p.u., respectively. The minimum magnitude observed for the downward ramps detected using the SDA for the various values of epsilon is 0.20 p.u., respectively. Furthermore, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, 75% of the downward ramps have a magnitude less than or equal to 0.3083 p.u., 0.3083 p.u., 0.3036 p.u., 0.2818 p.u. and 0.2891 p.u.

Therefore, based on Fig. 18 and Fig. 19 as well as the information mentioned above, it is evident that the downward ramps detected by the SDA generally have a slightly smaller magnitude than the upward ramps detected by the SDA. Additionally, if the size of epsilon decreases, the magnitude of the detected upward ramps also decreases. This is consistent with the trade-off introduced by epsilon. Conversely, the magnitude of the detected downward ramps seemingly increases if the size of epsilon decreases.

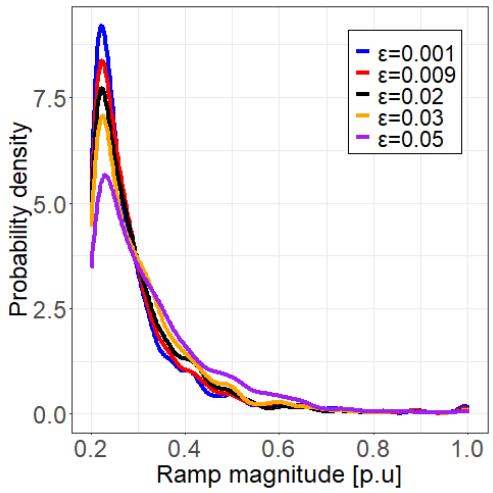


Fig. 18: Probability density functions of the ramp magnitude of upward ramps detected by the SDA for various parameter values.

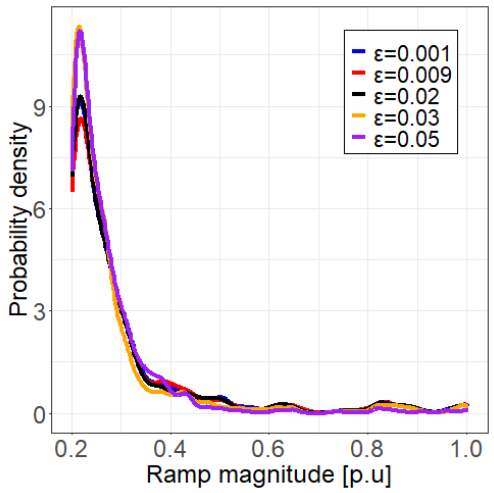


Fig. 19: Probability density functions of the ramp magnitude of downward ramps detected by the SDA for various parameter values.

3.4 L1-ramp detect with sliding window

3.4.1 Implementation

Sevlian and Rajagopal [21] proposed the L1-Ramp Detect with Sliding Window (L1-SW) to identify wind power ramp events in a large time series. The notion behind the detection algorithm is that the detected ramps correspond to optimal intervals of the wind power time series which satisfy the ramp rules. An optimal ramp interval is defined as the longest sequence of points that satisfy the ramp rule. It is, therefore, necessary to test all possible intervals of the wind power time series to ensure optimality [21]. Accordingly, the method uses a family of scoring functions that are related to a user defined ramp rule, and a dynamic programming recursion technique is then used to extract all significant ramp events.

A flowchart of the complete detection algorithm is shown in Fig. 20, and a detailed description is provided below.

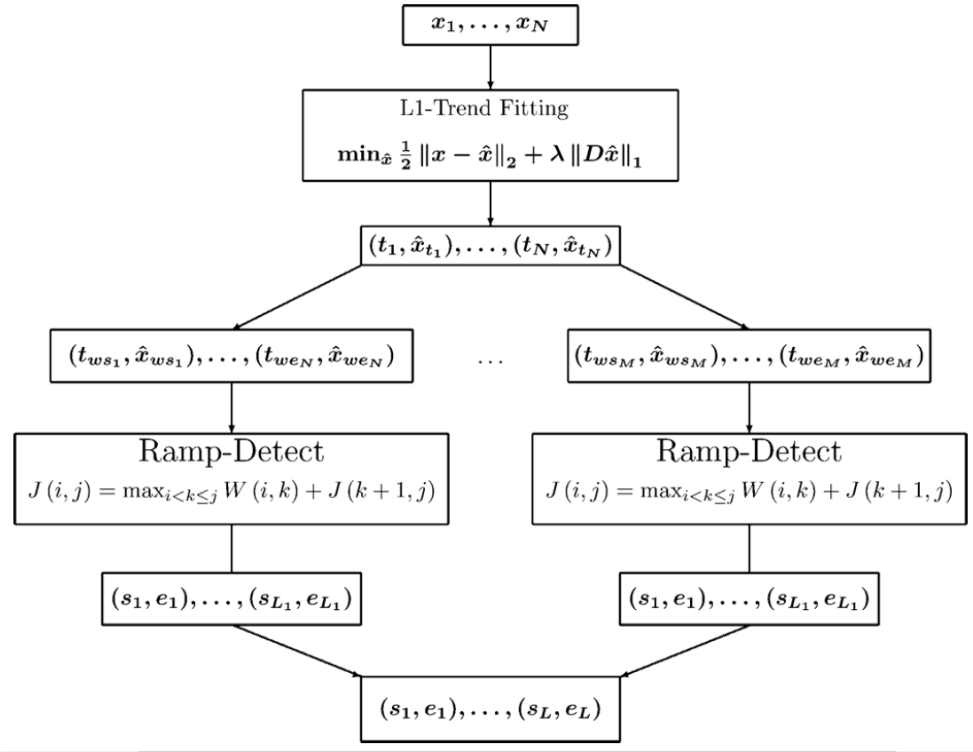


Fig. 20: Workflow of ramp detection using L1-SW [21].

Firstly, a piecewise linear trend fitting technique is applied to the data as a preprocessing step to remove insignificant noise and emphasise appropriate trends in the data, as well as to reduce the size of the data. The technique is implemented by first solving the convex problem given by

$$\min_{\hat{x}} \frac{1}{2} \|x - \hat{x}\|_2 + \lambda \|D\hat{x}\|_1 , \tag{85}$$

where x represents the signal, i.e. $x = \{p_1, \ldots, p_n\}$, and $t_i = i$. The D represents the second derivative operator, and $\|D\hat{x}\|$ imposes sparsity in the second derivative which results in $\hat{x}$ comprising a piecewise linear approximation [21]. The parameter λ is tunable and defines what is considered as the appropriate trends in the data that must be emphasised. Consequently, λ will affect which ramps will be detected. The parameter λ has a practical interpretation, namely it is associated with the shortest change that is allowed for the ramp rate. Accordingly, in power system management λ is associated with the smallest analysis intervals present in a specific dispatch study [21].

After the convex program in (85) is solved, it is necessary to extract the set of piecewise lines from $\hat{x}$. This can be achieved by setting a threshold γ, that must be carefully chosen, for the second derivative as formulated by

$$\|D\hat{x}\| > \gamma . \tag{86}$$

This results in the piecewise signal $\hat{X}_{PW}$ given by

$$\hat{X}_{PW} = \{(t_1, \hat{p}_{t1}), \ldots, (t_T, \hat{p}_{tT}) , \tag{87}$$

where $\hat{X}_{PW}$ includes only the points that complies with (86).

Sevlian and Rajagopal [21] proposed a sliding window approximate technique to support the detection of ramps for input time series with arbitrarily long duration by making it more manageable. It is possible to greatly reduce the computation time of the algorithm by splitting the time series into overlapping sections that will be processed separately or in parallel, and subsequently merging the detected ramps of each section in a post-processing step. Furthermore, it is required that the window length exceeds the duration of the longest ramp, so that the procedure is able to detect all ramps.

The resulting piecewise signal $\hat{X}_{PW}$ can be split up into overlapping windows, by defining an appropriate window length WL and window overlap WO. Accordingly, the start and end of the corresponding windows can be found using

$$ws_i = (i-1)(WL - WO) \tag{88}$$

and

$$we_i = (i)WL - (i-1)WO\,, \tag{89}$$

respectively. This will result in M overlapping windows, where M can be calculated using

$$M = \frac{N+WL-WO}{WL-WO} - 1\,. \tag{90}$$

Thereafter, a dynamic programming recursion can be applied to each of the M overlapping windows to detect the set of ramps within each window. The dynamic programming recursion technique maximises the objective function J and consequently extract all the significant ramp events according to the formulation

$$J(i,j) = \max_{i<k\leq j} W(i,k) + J(k.j)\,, \tag{91}$$

where $W(i,k)$ denotes the scoring function. Essentially, (91) entails assigning a score to every interval that satisfies the ramp rules, and the score is an increasing function of the length of the interval. Therefore, when the score is maximised, the set of optimal ramp intervals will be found.

Implementing the dynamic recursion function results in the optimal solution of the objective function J^*, for a given a scoring function, as given by

$$J^*(i,j) \;=\; \sum_{l=1}^{L^*} W(s_l^*\,, e_l^*)\,. \tag{92}$$

Here L^* represents the number of ramp events found in interval (i,j), and (s_l^*, e_l^*) denotes the indices corresponding to the start and end of the l^{th} ramp event, respectively. Sevlian and Rajagopal [21] present the code for the algorithm to solve the dynamic recursion technique.

It is required that the scoring function that evaluates the cost of each subsequence satisfies the super additivity property given by

$$\begin{aligned} &\forall i \;<\; k \;<\; j : \; R(i,j) = \; 1\,, \\ &W\;(i,j) > \; W(i,k) + W\;(k \;+\; 1, j)\,, \end{aligned} \tag{93}$$

to ensure optimal ramp detection. The scoring function must score ramp events and penalise non-ramp events.

An entire family of weight functions can satisfy the above conditions, nevertheless one possibility is given by

$$W(i,j) = (j - i)^2 \mathbf{1}\{R(i,j) = 1\}. \quad (94)$$

Other appropriate weight functions can also be used.

Finally, the results of each window are combined to ensure that the longest ramps will be found across the overlapping region. This will also provide the optimal set of ramps, which concludes the algorithm.

The advantage of this method is that it ensures that all ramps present in the wind power time series that satisfy the ramp rules are found, as well as that the detected ramps correspond to the longest sequence of points that satisfy the ramp rule, since all possible intervals are tested [21]. The L1-SW allows complicated rules to be easily considered, contrary to ad hoc techniques [21]. Additionally, the L1-SW is simple and easy to implement, and it produces consistent results.

3.4.2 Ramp detection results

The results of the L1-ramp detect with Sliding Window (L1-SW) are only shown for the first 400 discrete samples, to illustrate the effect of varying λ and γ. Fig. 21 to Fig. 24 depict the piecewise lines and the significant ramps extracted via the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$, respectively and the selected ramp definition. Fig. 25 to Fig. 27 depict the sparse second derivative resulting from the L1-trend filtering technique for $\lambda = 0.02, 0.5,$ and 1. It is recognised that the linear trend fitting technique greatly reduces the size of the data set that will be optimised via the dynamic recursion technique, and the dynamic recursion technique is capable of merging the piecewise lines that are adjacent to one another and have the same direction. The L1-SW identifies the longest sequence of points that satisfy the ramp rule in accordance with the emphasised trends.

For an appropriate value of γ, it is evident that as the value of λ decreases, the number of line segments increases and correspondingly the magnitude and duration of the segments also decreases, which ensures a more accurate approximation of the signal. Conversely, as the value of λ increases, the number of line segments decreases and correspondingly the magnitude and duration of the segments also increases, resulting in a less accurate approximation of the signal. Thus, λ introduces a trade-off between accuracy and the number of piecewise lines. To provide more insight, the L1-trend filtering technique is used to estimate the underlying trends in the signal, and λ defines the trends that must be emphasised. Specifically, λ is associated with the shortest permissible changes in the ramp rate. Accordingly, if the value of λ decreases, more abrupt changes in the ramp rate are allowed, resulting in the second derivative of the filtered signal including more values, as well as higher values, as shown in Fig. 25 to Fig. 27. Thus, when thresholding the second derivative with a constant γ to extract the piecewise line segments, a smaller value of λ will result in more values of the second derivative being above the threshold when compared to a larger value of λ, and consequently more piecewise lines will be extracted. Furthermore, for a given value of λ, it is evident that a smaller value of γ results in the extraction of more piecewise lines when compared to a larger value of γ, as illustrated in Fig. 22 and Fig. 23, since more points will be above the threshold. The choice of γ is important to ensure the extraction of appropriate line segments.

The detected ramps depend on the piecewise lines extracted via the linear-trend filtering technique, therefore, λ and γ affect the detected ramps. For smaller values of λ and an appropriate value of γ, the

piecewise lines will fit the signal more accurately. Less ramp variations are, therefore, part of each segment, and separate segments are rather used to approximate these variations. The segments approximating the ramp variations may cause the dynamic programming recursion to break. Consequently, using the L1-SW for smaller values of λ results in the detection of smaller ramps with shorter duration that fits the signal more accurately, when compared to using the L1-SW for greater values of λ. It is, furthermore, recognised that the piecewise lines and the corresponding ramps are not anchored to the points of the signal, since it is anchored to the filtered signal.

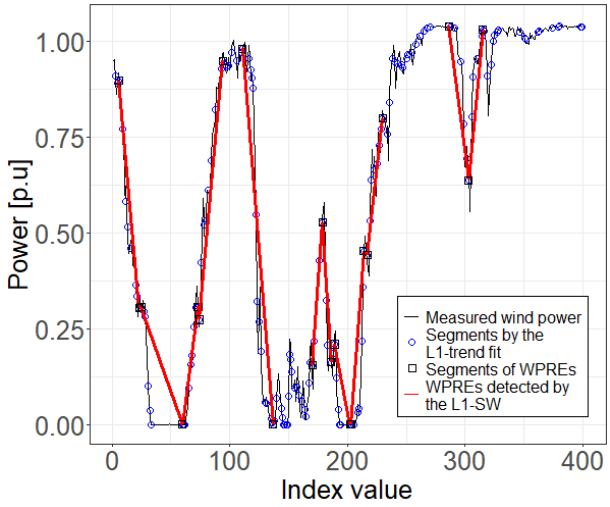


Fig. 21: Results of the ramp detection via the L1-SW for $\lambda = 0.02,\ \gamma = 5 \times 10^{-6}$ *and* $P_{val} = 0.2$.

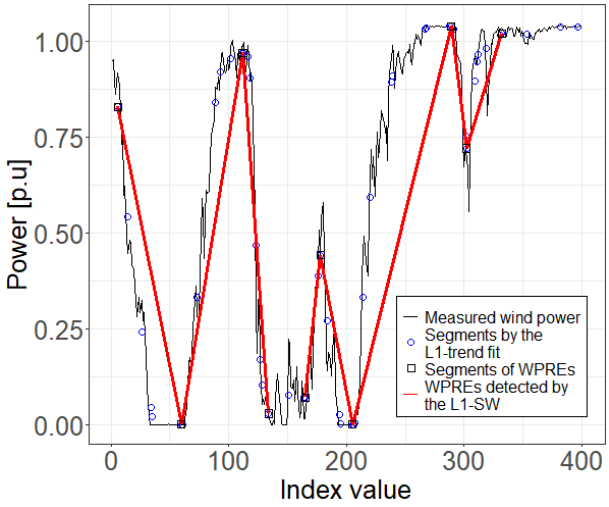


Fig. 22: Results of the ramp detection via the L1-SW for $\lambda = 0.5, \gamma = 1 \times 10^{-4}$ *and* $P_{val} = 0.2$.

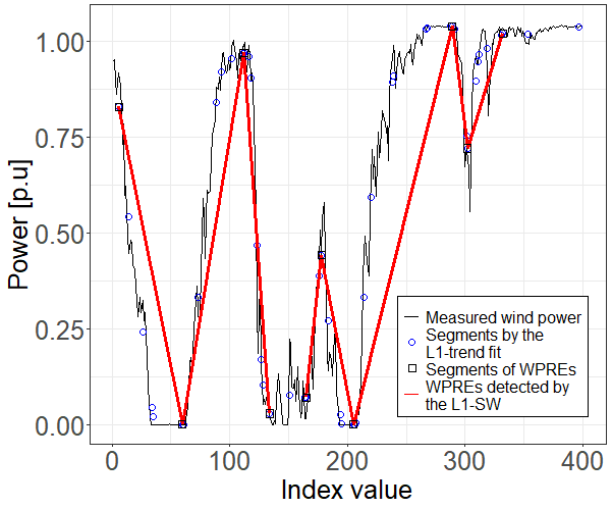


Fig. 23: Results of the ramp detection via the L1-SW for $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ *and* $P_{val} = 0.2$.

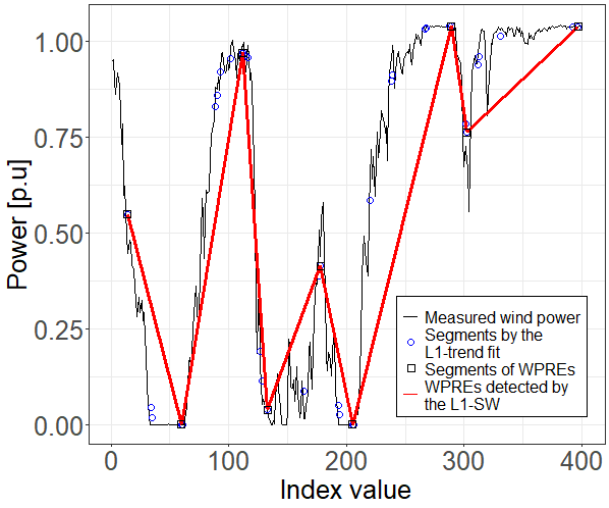


Fig. 24: Results of the ramp detection via the L1-SW for $\lambda = 1, \gamma = 1 \times 10^{-4}$ *and* $P_{val} = 0.2$.

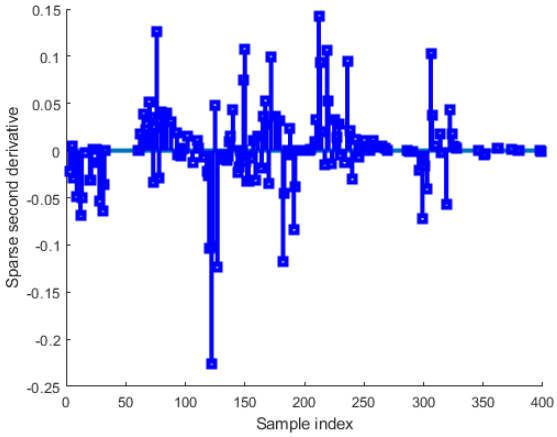


Fig. 25: The sparse second derivative resulting from the L1-trend filtering technique for $\lambda = 0.02$.

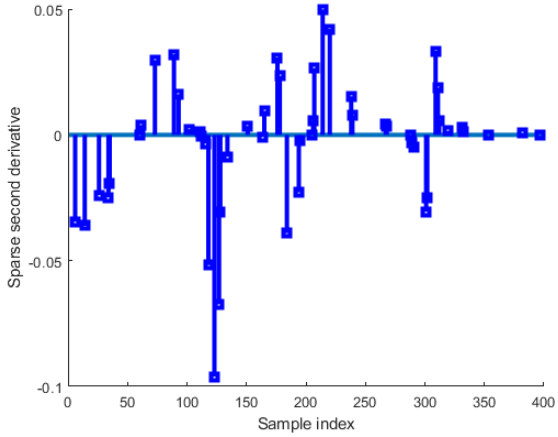


Fig. 26: The sparse second derivative resulting from the L1-trend filtering technique for $\lambda = 0.5$.

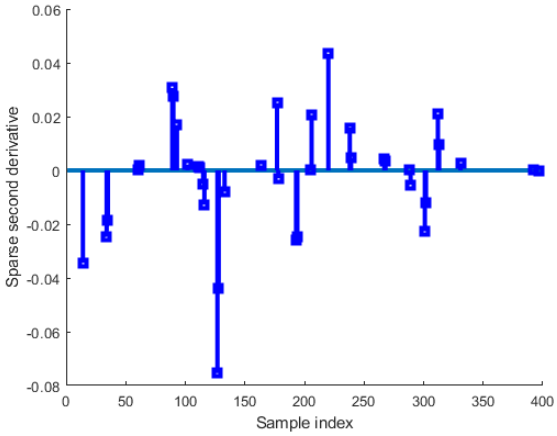


Fig. 27: The sparse second derivative resulting from the L1-trend filtering technique for $\lambda = 1$.

3.4.3 Statistical analysis of key ramp features

3.4.3.1 Ramp duration

Fig. 28 and Fig. 29 depict the Probability Density Functions (PDFs) of the ramp duration of the upward and downward ramps, respectively, detected using the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$. It is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the L1-SW will have a shorter duration rather than a longer duration.

The probability density functions for the upward and downward ramps display a high degree of similarity. The upward ramps detected by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$ have a mean ramp duration of 147.76 min, 327.59 min, 333.10 min and 380.01 min, respectively. For $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$, the maximum ramp duration observed for the upward ramps is 1130 min, 1370 min, 1400 min and 1460 min, respectively. The minimum duration observed for the

upward ramps detected using the L1-SW for the various values of λ and γ is 10 min, 40 min, 40 min, and 60 min, respectively. Furthermore, for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$, 75% of the upward ramps have a duration less than or equal to 190 min, 440 min, 440 min and 500 min, respectively. Conversely, the downward ramps detected by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$ have a mean ramp duration of 158.11 min, 372.64 min, 370.27 min and 440.53 min, respectively. Additionally, for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$, the maximum ramp duration observed for the downward ramps is 1230 min, 1830 min, 1840 min and 1820 min, respectively. The minimum duration observed for the downward ramps detected using the L1-SW for the various values of λ and γ is 10 min, 50 min, 50 min and 70 min, respectively. Furthermore, for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$, respectively, 75% of the downward ramps have a duration less than or equal to 200 min, 490 min, 480 min and 580 min.

Therefore, based on Fig. 28 and Fig. 29 as well as the information mentioned above, it is evident that the downward ramps detected by the L1-SW generally have a longer duration than the upward ramps detected by the L1-SW. Additionally, if λ decreases, the duration of the detected upward and downward ramps also decreases, however, the effect of γ is not as evident. Nevertheless, it is important to exercise the necessary caution when choosing the γ value, since it produces the thresholded piecewise signal.

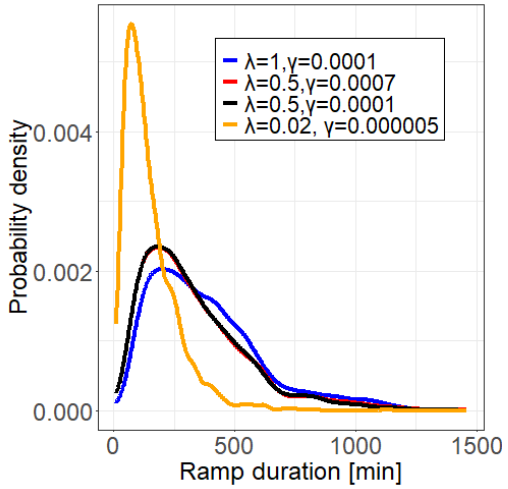


Fig. 28: Probability density functions of the ramp duration of upward ramps detected by the L1-SW for various values of λ and γ.

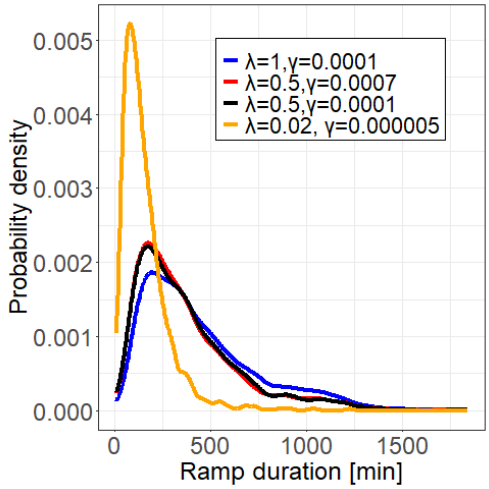


Fig. 29: Probability density functions of the ramp duration of downward ramps detected by the L1-SW for various values of λ and γ.

3.4.3.2 Ramp magnitude

Fig. 30 and Fig. 31 depict the Probability Density Functions (PDFs) of the ramp magnitude of the upward and downward ramps, respectively, detected using the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$. It is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the L1-SW will have a small magnitude rather than a large magnitude.

The probability density functions for the upward and downward ramps display a high degree of similarity. The upward ramps detected by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda =$

$0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$ have a mean ramp magnitude of 0.4626 p.u., 0.5715 p.u., 0.5713 p.u. and 0.5924 p.u., respectively. The maximum magnitude observed for the upward ramps detected using the L1-SW for the various values of λ and γ is 1.0433 p.u., 1.0409 p.u., 1.0409 p.u. and 1.0409 p.u., respectively. The minimum magnitude observed for the upward ramps detected using the L1-SW for the various values of λ and γ is 0.20 p.u., respectively. Furthermore, for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$, respectively, 75% of the upward ramps have a magnitude less than or equal to 0.6057 p.u., 0.8047 p.u., 0.8029 p.u. and 0.8383 p.u. Conversely, the downward ramps detected by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$ have a mean ramp magnitude of 0.4507 p.u., 0.5660 p.u., 0.5660 p.u. and 0.5809 p.u., respectively. The maximum magnitude observed for the downward ramps detected using the L1-SW for the various values of λ and γ is 1.0406 p.u., 1.0395 p.u., 1.0391 p.u. and 1.0382 p.u., respectively. The minimum magnitude observed for the downward ramps detected using the L1-SW for the various values of λ and γ is 0.20 p.u., respectively. Furthermore, for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$; $\lambda = 0.5, \gamma = 1 \times 10^{-4}$; $\lambda = 0.5, \gamma = 7 \times 10^{-4}$ and $\lambda = 1, \gamma = 1 \times 10^{-4}$, respectively, 75% of the downward ramps have a magnitude less than or equal to 0.5815 p.u., 0.7881 p.u., 0.7885 p.u. and 0.8261 p.u.

Therefore, based on Fig. 30 and Fig. 31 as well as the information mentioned above, it is evident that the downward ramps detected by the L1-SW generally have a slightly smaller magnitude than the upward ramps detected by the L1-SW. Additionally, if λ decreases, the magnitude of the detected upward and downward ramps also decreases, however, the effect of γ is not as evident. Nevertheless, it is important to exercise the necessary caution when choosing the γ value, since it produces the thresholded piecewise signal.

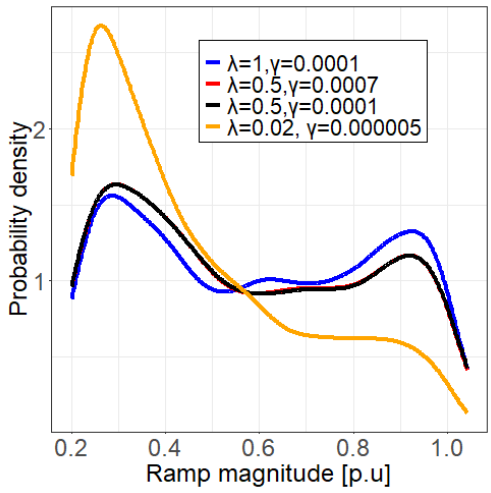


Fig. 30: Probability density functions of the ramp magnitude of upward ramps detected by the L1-SW for various values of λ and γ.

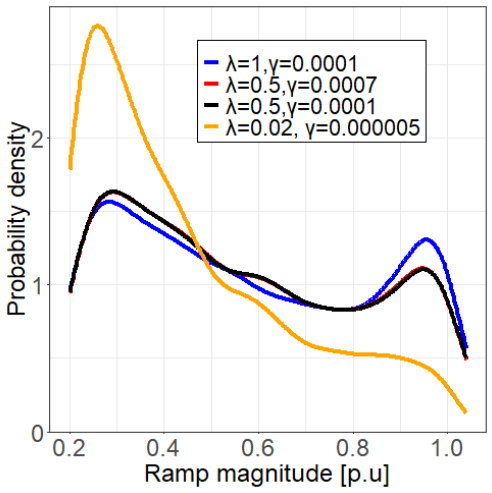


Fig. 31: Probability density functions of the ramp magnitude of downward ramps detected by the L1-SW for various values of λ and γ.

3.5 Optimised swinging door algorithm

3.5.1 Implementation

Zhang *et al*. [30] developed the Optimised Swinging Door Algorithm (OpSDA) with the objective of minimizing the number of individual ramp segments that is used to linearly approximate a signal, as obtained from the SDA. Accordingly, segments that are adjacent to one another and have the same direction can be merged. For this purpose, a dynamic programming algorithm, adopted from Sevlian and Rajagopal [21], is used to optimise the segments obtained from the swinging door algorithm. Dynamic programming entails breaking a complex problem up into a collection of simpler subproblems to make it easier to solve [30], [54].

The overall process of the OpSDA is illustrated in Fig. 32, and is summarised as follows [30]. Firstly, the measured wind power data is segmented using the SDA with a predefined epsilon. A detailed description of the SDA is provided in section 3.3. The resulting segments will then be used as an input to the optimizing process, which includes the sliding window approximate technique and the dynamic programming recursion from Sevlian and Rajagopal [21], to yield the set of optimised significant ramps. The overall process of the L1-SW and the OpSDA is similar in the sense that in the overall process the piecewise linear trend fitting technique is replaced with the original swinging door algorithm as the segmenting process. Therefore, it is redundant to include the sliding window approximate technique and dynamic programming recursion, as it can be found in section 3.3 and section 3.4, respectively.

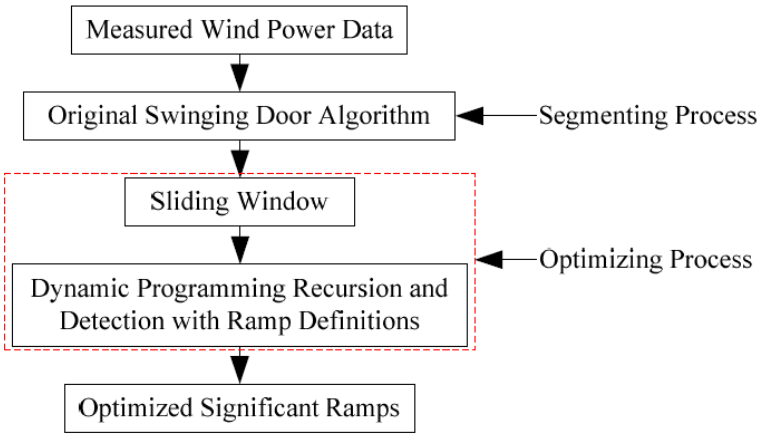


Fig. 32: The overall process of the OpSDA [30].

It is concluded that the OpSDA performs significantly better compared to the original SDA, as well as similarly or better compared to the L1-SW, while also being more computationally inexpensive. It is recognised that the detected ramps from the SDA and OpSDA is anchored to measurement points of the wind power signal, whereas the detected ramps from the L1-SW is anchored to the points of the filtered signal. Furthermore, a framework was developed to determine the optimal value of the tunable parameter in the SDA based on using the OpSDA as a baseline, since in some cases it may be preferable to employ the SDA because of its computational expedience.

3.5.2 Ramp detection results

The results of the OpSDA are only shown for the first 400 discrete samples, to illustrate the effect of varying epsilon. Fig. 33 to Fig. 37 depict the segments extracted via the SDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, as well as the significant ramps satisfying the selected ramp definition that were extracted via the OpSDA by optimizing these segments via the dynamic programming recursion. It is evident that the dynamic programming recursion is capable of merging the segments that are adjacent to one another which have the same direction. The ramps extracted via the OpSDA, therefore, have a larger magnitude and duration than the ramps extracted via the SDA, and the number of ramps also increased.

The significant ramps detected via the OpSDA depend on the segments extracted via the SDA. For smaller values of epsilon, the segments fit the signal more accurately. Less ramp variations are consequently part of each segment, and separate segments are rather used to approximate these variations. The segments approximating the ramp variations may cause the dynamic programming recursion to break. Therefore, using smaller values of epsilon results in the detection of smaller ramps with shorter duration that fits the signal more accurately, when compared to using the OpSDA for greater values of epsilon. Furthermore, since the SDA fails to capture the peaks of the signal, i.e. the SDA does not successfully extract the start and end-point of the ramps, in turn the OpSDA also fails to capture the peaks of the signal.

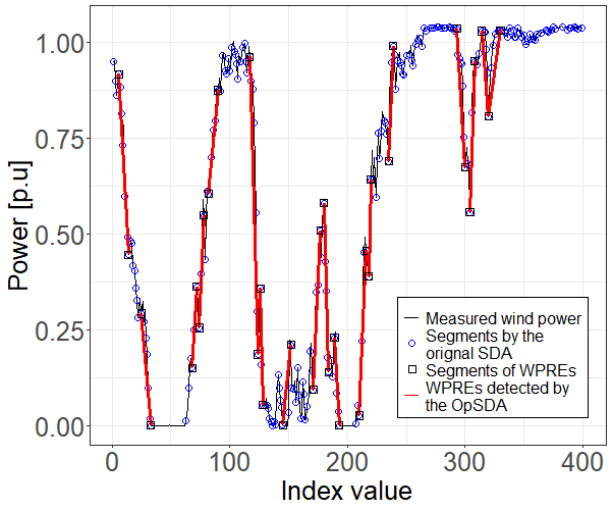


Fig. 33: Results of the ramp detection via the OpSDA for $\varepsilon = 0.001$ and $P_{val} = 0.2$.

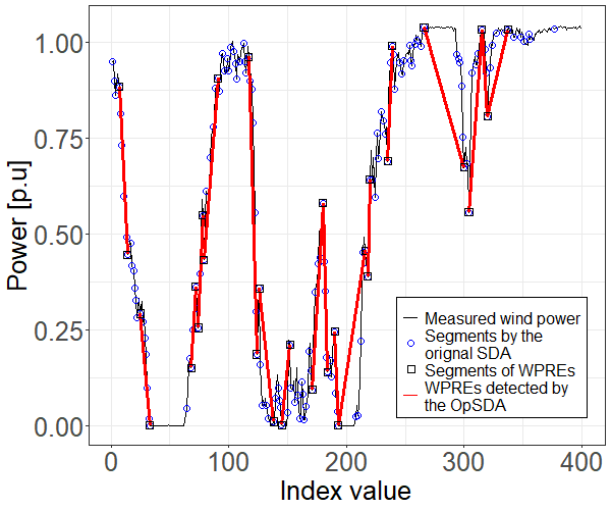


Fig. 34: Results of the ramp detection via the OpSDA for $\varepsilon = 0.009$ and $P_{val} = 0.2$.

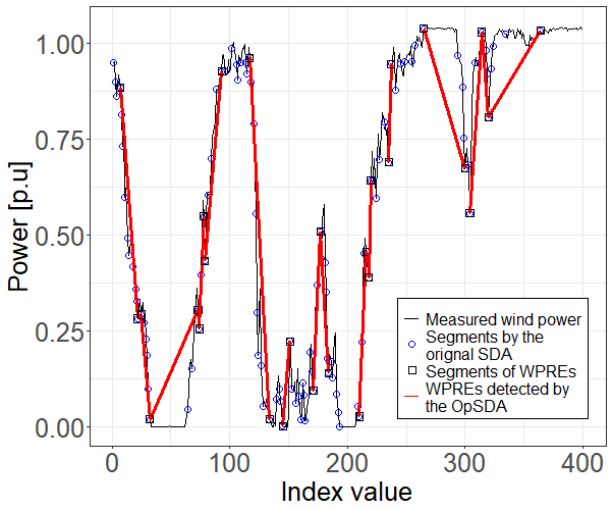


Fig. 35: Results of the ramp detection via the OpSDA for $\varepsilon = 0.02$ and $P_{val} = 0.2$.

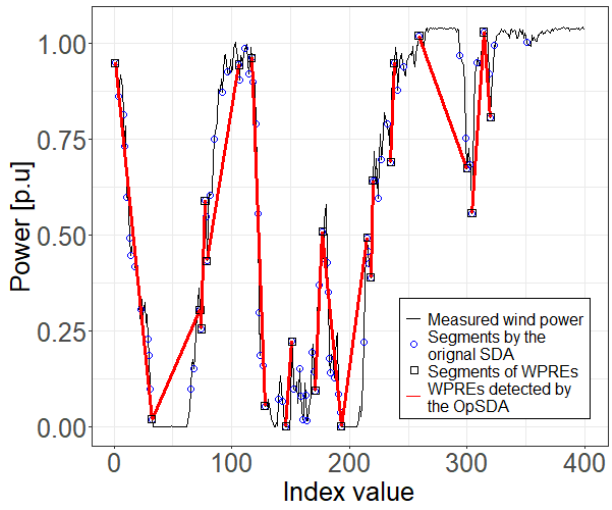


Fig. 36: Results of the ramp detection via the OpSDA for $\varepsilon = 0.03$ and $P_{val} = 0.2$.

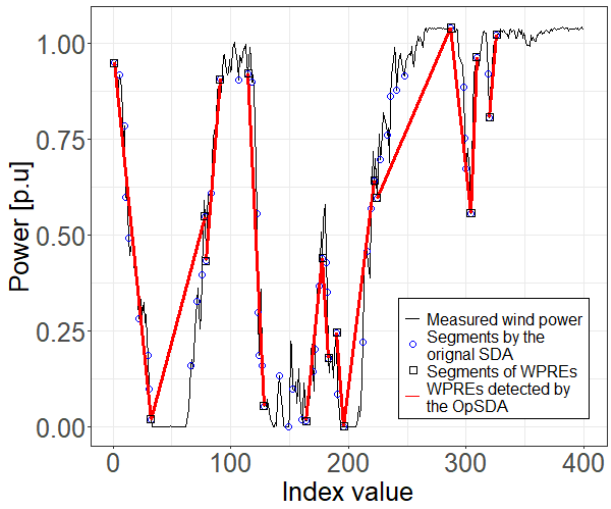


Fig. 37: Results of the ramp detection via the OpSDA for $\varepsilon = 0.05$ and $P_{val} = 0.2$.

3.5.3 Statistical analysis of key ramp features

3.5.3.1 Ramp duration

Fig. 38 and Fig. 39 depict the Probability Density Functions (PDFs) of the ramp duration of the upward and downward ramps, respectively, detected using the OpSDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%. It is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the OpSDA will have a short duration rather than a long duration.

The PDFs for the upward ramps are approximately similar to the PDFs for the downward ramps. The upward ramps detected by the OpSDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% have a mean ramp duration of 84.84 min, 112.32 min, 154.15 min, 193.68 min and 261.88 min, respectively. For $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, the maximum duration observed for the upward ramps is 1090 min, 1180

min, 1420 min, 2040 min and 2180 min, respectively. The minimum duration observed for the upward ramps detected using the OpSDA for the various values of epsilon is 20 minutes, respectively. Furthermore, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, 75% of the upward ramps have a duration less than or equal to 100 min, 130 min, 180 min, 240 min and 340 min. Conversely, the downward ramps detected by the OpSDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% have a mean ramp duration of 87.72 min, 104.83 min, 139.31 min, 172.79 min and 231.95 min, respectively. Additionally, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, the maximum duration observed for the downward ramps is 450 min, 1050 min, 1320 min, 1370 min and 1930 min, respectively. The minimum duration observed for the downward ramps detected using the OpSDA for the various values of epsilon is 10 minutes, respectively. Furthermore, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, 75% of the downward ramps have a duration less than or equal to 110 min, 130 min, 170 min, 220 min and 310 min.

Therefore, based on Fig. 38 and Fig. 39 as well as the information mentioned above, it is evident that the downward ramps detected by the OpSDA generally have a shorter duration than the upward ramps detected by the OpSDA. Additionally, if the size of epsilon decreases, the duration of the detected upward and downward ramps also decreases.

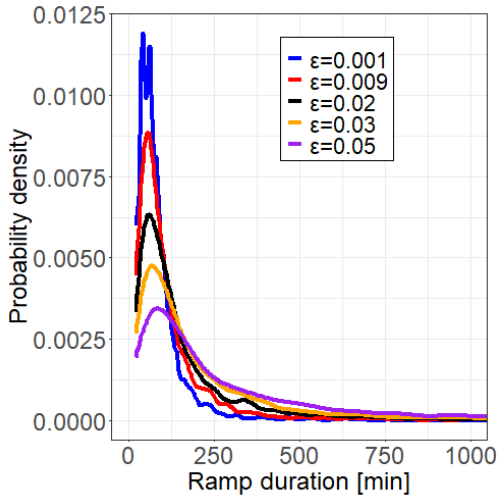


Fig. 38: Probability density functions of the ramp duration of upward ramps detected by the OpSDA for various values of epsilon.

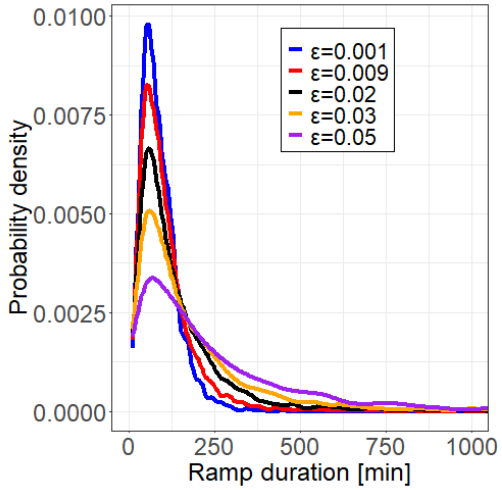


Fig. 39: Probability density functions of the ramp duration of downward ramps detected by the OpSDA for various values of epsilon.

3.5.3.2 Ramp magnitude

Fig. 40 and Fig. 41 depict the Probability Density Functions (PDFs) of the ramp magnitude of the upward and downward ramps, respectively, detected using the OpSDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%. It is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the OpSDA will have a small magnitude rather than a large magnitude.

The upward ramps detected by the OpSDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% have a mean ramp magnitude of 0.3996 p.u., 0.4112 p.u., 0.4294 p.u., 0.4526 p.u. and 0.4819 p.u., respectively. The maximum magnitude observed for the upward ramps detected using the OpSDA for the various values of epsilon is 1.0433 p.u., 1.040 p.u., 1.0368 p.u., 1.0401 p.u. and 1.0433 p.u., respectively. The minimum magnitude observed for the upward ramps detected using the OpSDA for the various values of epsilon is 0.20 p.u., respectively. Furthermore, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, 75% of the

upward ramps have a magnitude less than or equal to 0.4922 p.u., 0.5066 p.u., 0.5356 p.u., 0.5895 p.u. and 0.6461 p.u.

Conversely, the downward ramps detected by the OpSDA for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5% have a mean ramp magnitude of 0.4063 p.u., 0.4146 p.u., 0.4316 p.u., 0.4521 p.u. and 0.4831 p.u., respectively. The maximum magnitude observed for the downward ramps detected using the OpSDA for the various values of epsilon is 1.0379 p.u., 1.040 p.u., 1.0356 p.u., 1.0390 p.u. and 1.0378 p.u., respectively. The minimum magnitude observed for the downward ramps detected using the OpSDA for the various values of epsilon is 0.20 p.u., respectively. Furthermore, for $\varepsilon = 0.1\%, 0.9\%, 2\%, 3\%$ and 5%, respectively, 75% of the downward ramps have a magnitude less than or equal to 0.5029 p.u., 0.5107 p.u., 0.5489 p.u., 0.5832 p.u. and 0.6435 p.u.

Therefore, based on Fig. 40 and Fig. 41 as well as the information mentioned above, it is evident that magnitude of the upward and downward ramps detected by the OpSDA are approximately similar. Additionally, if the size of epsilon decreases, the magnitude of the detected upward and downward ramps also decreases.

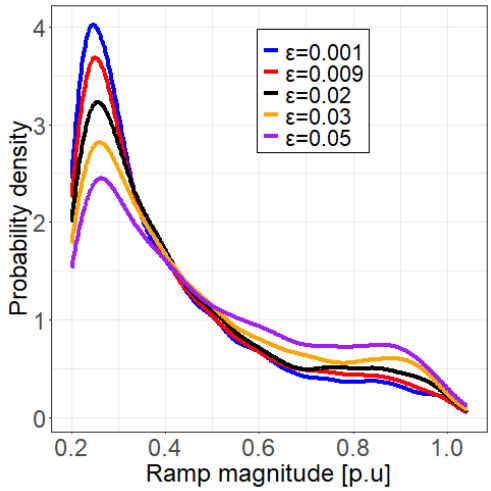


Fig. 40: Probability density functions of the ramp magnitude of upward ramps detected by the OpSDA for various values of epsilon.

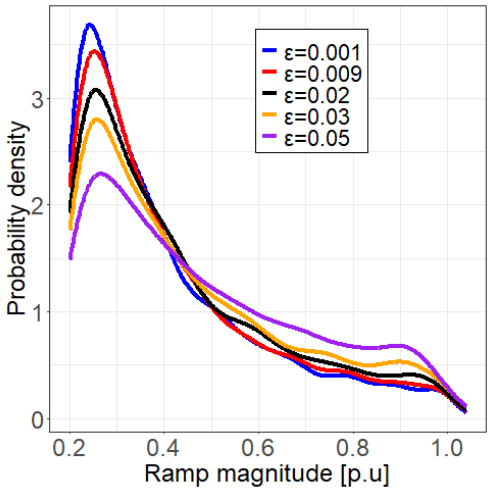


Fig. 41: Probability density functions of the ramp magnitude of downward ramps detected by the OpSDA for various values of epsilon.

3.6 A multi-parameter segmentation algorithm for wind power ramp detection

3.6.1 Implementation

In this section, an original algorithm is proposed to decompose a temporal signal into upward and downward ramping segments. Significant wind power ramp events are subsequently identified from the derived segments according to a user-specified definition of a significant ramp.

A ramp event is defined as a significant increase or decrease in wind power over a limited period of time. The proposed algorithm is consequently designed to deconstruct a temporal wind power series into increasing or decreasing ramps. The proposed algorithm should be capable of detecting all ramps of varying duration which satisfy some minimum magnitude threshold [21]. Computational expedience is, furthermore, considered imperative.

The proposed model is adapted from (4) by incorporating a tuneable offset parameter γ that is defined in per unit power instead of the multiplying factor β. The initial power profile, P, is given by the set

$$P = \{p(t_k), \quad k = 1, 2, 3 \cdots N\}, \tag{95}$$

where N denotes the number of time instants in the profile. The profile is normalized to yield the normalized profile P_n, given by

$$P_n = \left\{ p_n(t_k) \middle| \; p_n(t_k) = \frac{p(t_k)}{P_{Ref}}, k = 1,2,3 \cdots N \right\}, \tag{96}$$

where P_{Ref} denotes the normalization factor, which can be defined as the nominal capacity or the maximum wind power in profile P. The condition for an upward ramp event, $R_+(t_i, t_j)$, is defined by the relationship

$$R_+(t_i, t_j) = \prod_{m=i}^{j} \mathbf{1}_{\{p_n(t_m) > (max(p_n(t_i),\ldots,p_n(t_m)) - \gamma), \;\; 0 < \gamma < 1\}}. \tag{97}$$

The offset parameter defines the permitted vertical fluctuation in a given ramp segment. The offset parameter fixes the magnitude of the permissible fluctuation in terms of the maximum value contained within the entire power profile, rather than a value associated with the iterative interval from t_i to t_m. This amendment can significantly impact the results obtained during the initial segmentation process.

Conversely, the segmentation rule for downward ramps, $R_-(t_i, t_j)$, is defined as

$$R_-(t_i, t_j) = \prod_{m=i}^{j} \mathbf{1}_{\{p_n(t_m) < (min(p_n(t_i),\ldots,p_n(t_m)) + \gamma), 0 < \gamma < 1\}}. \tag{98}$$

The direction, duration, and magnitude of each upward and downward ramp occurrence is initially unknown. The proposed algorithm is implemented by iteratively extending the end-point defined by j from a starting point defined by i. $R_+(t_i, t_j)$ and $R_-(t_i, t_j)$ are calculated for the segment defined by t_i and t_j using (97) and (98), respectively. The end of a ramp is defined as the time instant t_j before the applicable segmentation rule, i.e. $R_+(t_i, t_j)$ or $R_-(t_i, t_j)$ equals 0. In the case of an upward ramp, for example, $R_+(t_i, t_j)$ equals 0 when the trend violates the condition defined in (97), i.e. when $p_n(t_j) \leq (max\left(p_n(t_i), \ldots, p_n(t_j)\right) - \gamma)$. Additionally, the end-point must be updated to ensure that the accurate ramp start and ramp end is detected. For upward ramps, the end-point of the segment is selected as the index with the largest magnitude. Similarly, for a downward ramp, the end-point of the segment is selected as the index with the smallest magnitude. An end-point in turn becomes the starting point of the following segment. This process continues until all samples in the wind power time series are evaluated. The power profile for a ramp segment can be expressed by the relationship

$$P_s(t_i, t_j) = \left\{ \begin{matrix} p_s(t_m), t_i \leq t_m \leq t_j, \\ (R_-(t_i, t_j) = 1) \vee (R_+(t_i, t_j) = 1) \end{matrix} \right\}, \tag{99}$$

where p_s denotes the initial power at time instant t_m.

Fig. 42 (Left) shows an illustrative example of the proposed segmentation rule. The application of (98) yields a well-defined downward ramp from i to j. It is, however, clear that the extracted ramp is interjected by a significant horizontal segment. A post-processing algorithm is applied in order to discard horizontal segments based on a tunable parameter φ. This parameter specifies the minimum length of the horizontal segment to be retained. A segment that remains within γ for a duration longer than φ is reclassified as a horizontal segment. Fig. 42 (Right) shows the results of the post-processing step which is formally defined below.

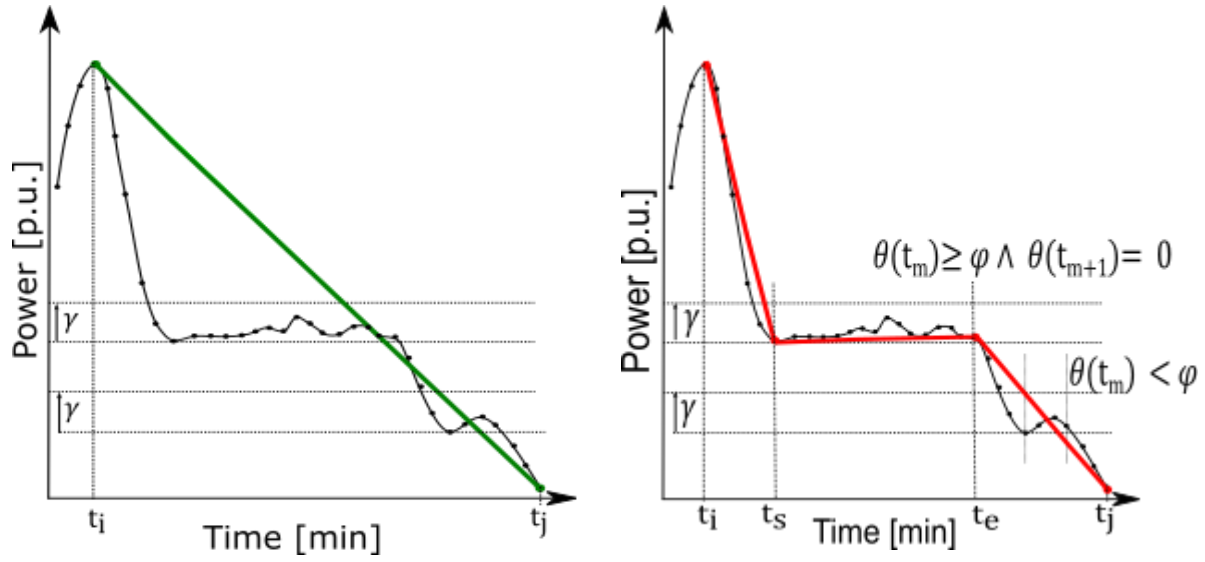


Fig. 42: (Left) Ramping segments identified using the proposed model parameterised by γ only (green), compared to (Right) the identified ramping segments when a horizontal translation term is introduced (red). The scale and parameters, γ and φ, are arbitrary for explanation purposes.

For downward ramps, the first step in the post-processing algorithm is to extract a set that defines the minimum power values associated with the interval from t_i to t_m, for $t_i \leq t_m \leq t_j$. This yields the set P_{min}, where

$$P_{min} = \{\rho_{min}(t_m) \mid p_{min}(t_m) = min\big(p_s(t_i), \dots, p_s(t_m)\big), \quad \forall m = i, \dots, j\}. \tag{100}$$

For upward ramps, the first step in the post-processing algorithm is to extract a set that defines the maximum power values associated with the interval from t_i to t_m, for $t_i \leq t_m \leq t_j$. This yields the set P_{max}, where

$$P_{max} = \{\rho_{max}(t_m) \mid p_{max}(t_m) = max\big(p_s(t_i), \dots, p_s(t_m)\big), \quad \forall m = i, \dots, j\}. \tag{101}$$

The steps to detect horizontal segments in downward ramps and upward ramps are similar in the sense that ρ_{min} is replaced with ρ_{max} in the governing equations. The detection of horizontal segments is therefor only described for downward ramps.

Horizontal segments are subsequently identified using

$$\theta = \left\{ \theta(t_m) \left| \begin{array}{lll} \theta(t_m) = 0 & if & t_m = t_i \\ \theta(t_m) = 0 & if & \rho_{min}(t_m) \neq \rho_{min}(t_{m-1}) \\ \theta(t_m) = \theta(t_{m-1}) + 1 & if & \rho_{min}(t_m) = \rho_{min}(t_{m-1}) \\ \forall m = i, \dots, j & & \end{array} \right. \right\}, \quad (102)$$

where θ is an iterative counter which stores the number of consecutive samples where the power within the considered segment is greater than the previous minima. A horizontal segment is characterised by a start time, t_s, and an end time, t_e, which are derived as follows:

$$T_s = \left\{ t_s \mid \left(t_s = t_m - \theta(t_m)\right) if \begin{pmatrix} (\theta(t_m) \geq \varphi \ \wedge \ \theta(t_{m+1}) = 0) \\ \vee \\ (\theta(t_m) \geq \varphi \ \wedge \ m = j) \end{pmatrix}, \forall m = i, \dots, j \right\}, \quad (103)$$

and

$$T_e = \left\{ t_e \mid (t_e = t_m) \, if \begin{pmatrix} (\theta(t_m) \geq \varphi \ \wedge \ \theta(t_{m+1}) = 0) \\ \vee \\ (\theta(t_m) \geq \varphi \ \wedge \ m = j) \end{pmatrix}, \forall m = i, \dots, j \right\}. \quad (104)$$

The identified start- and end-points demarcate the horizontal segments contained in P_s.

Following the segmentation of the wind power signal using the proposed methodology, the user-specified definition of a significant ramp is applied to extract the wind power ramp events. The benefit of this segmentation algorithm is its structural and computational expedience as it only requires two tunable parameters. Additionally, (4) is already satisfied for the obtained segments and does not have to be retested. This algorithm includes all the optimisations contained in the OpSDA [54], namely merging adjacent segments with the same ramp direction, managing bumps (fluctuations) and insignificant ramps processing (horizontal segments).

There is no consensus in literature regarding the definitions for an insignificant fluctuation, ramp magnitude and ramp duration when considering actual/measured data. According to Florita *et al.* [16], two applications can be used to define an insignificant fluctuation. Firstly, the definition of an insignificant fluctuation is dependent on the accuracy of the measurement device, as characterised by the distribution of the measurement uncertainty. Secondly, it is dependent on the usage and importance of the measure, which is characterised by the economics of the power systems and its relative significance in driving operations. Neither of these applications are considered for the determination of the parameterisations.

3.6.2 Ramp detection results

The results of the multi-parameter segmentation algorithm are only shown for the first 400 discrete samples, to illustrate the effect of varying γ and φ. Fig. 43 to Fig. 48 depict the segments and significant ramps extracted via the multi-parameter segmentation algorithm for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5; \gamma = 0.01, \varphi = 2; \gamma = 0.01, \varphi = 5; \gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$ and the selected ramp definition. The multi-parameter segmentation algorithm is capable of decomposing the temporal wind power signal into increasing and decreasing segments to facilitate ramp detection.

If the value of γ decreases, correspondingly, the size of the permissible vertical fluctuation within a segment decreases. Therefore, more segments are used to approximate the signal since the threshold for the permissible vertical fluctuation is more easily violated, and less fluctuations are also present within the upward and downward segments. Accordingly, the magnitude and duration of the corresponding segments also decreases. This results in a more accurate approximation of the signal. When increasing γ, correspondingly, the size of the permissible vertical fluctuation increases. This results in the integration of more fluctuations, as well as larger fluctuations, in the upward and downward segments. Accordingly, the magnitude and duration of the corresponding segments also increases, and less segments are used to approximate the signal resulting in a less accurate approximation of the signal.

As mentioned above, φ is introduced to aid in the identification of horizontal segments. Specifically, a segment that remains within γ for a duration longer than φ is reclassified as a horizontal segment. For smaller values of γ, the effect of φ is not very evident since the segments do not include many fluctuations. For larger values of γ, the segments include more fluctuations and the effect of φ is more evident. Therefore, it is evident that for a larger value of γ more horizontal segments are present when compared to a smaller value of γ.

The duration and frequency of the horizontal segments depend on the value of φ. For a given value of γ, if φ decreases, more horizontal segments are found since segments that remain within γ for a duration longer than φ occur more often. It is recognised that the shortest duration observed for the horizontal segments decreases as φ decreases, noting that in turn the frequency of short horizontal segments also increases, since the duration of the fluctuation just has to be greater than or equal to φ in order to be reclassified as a horizontal segment. Therefore, it is evident that as φ decreases, the obtained segments tend to the segments extracted for a smaller γ, since short horizontal segments occur more frequently and these short horizontal segments resemble small ramps that splits the upward and downward segments, instead of forming part of the segments. Furthermore, since these fluctuating segments are no longer integrated into the upward and downward segments, the magnitude and duration of the extracted segments also decrease as shown clearly in Fig. 47 and Fig. 48. Conversely, as φ increases, the threshold for φ is not as easily violated. Thus, fewer horizontal segments are found, and the shortest duration observed for the horizontal segments also increase. Correspondingly, more fluctuations, as well as longer fluctuations, are included within the upward and downward segments. Therefore, the magnitude and duration of the extracted segments also increases.

The detected ramps correspond to the segments from the multi-parameter segmentation algorithm that satisfies the selected ramp definition. Accordingly, γ and φ affects the magnitude and duration of the detected ramps. It is evident that if γ and/or φ increases, more and larger significant ramps are detected due to more ramps with a large magnitude being extracted via the multi-parameter segmentation algorithm. Furthermore, it is recognised that the detected ramps correspond to visually detected ramps.

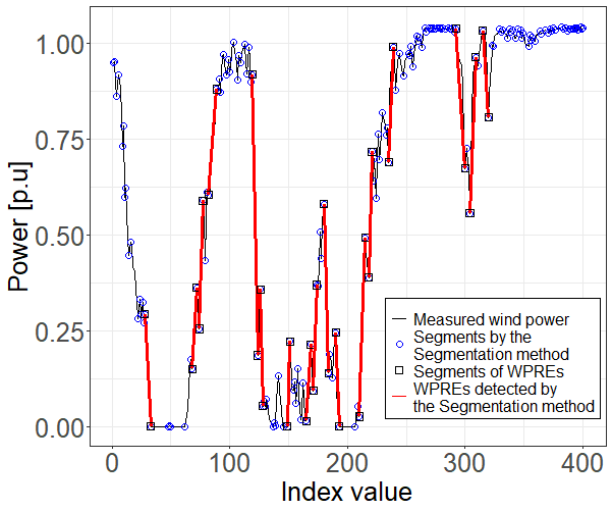


Fig. 43: Results of the ramp detection via the multi-parameter segmentation algorithm for $\gamma = 1 \times 10^{-4}, \varphi = 2$ and $P_{val} = 0.2$.

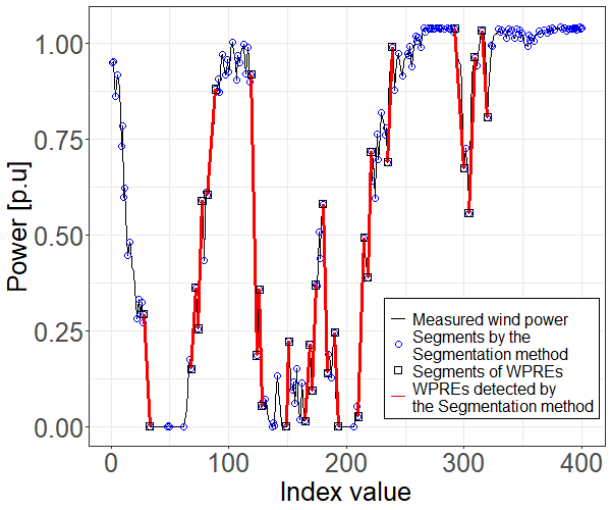


Fig. 44: Results of the ramp detection via the multi-parameter segmentation algorithm for $\gamma = 1 \times 10^{-4}, \varphi = 5$ and $P_{val} = 0.2$.

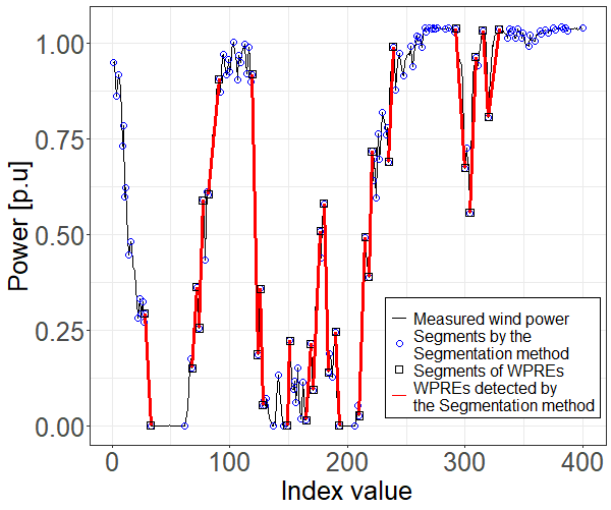


Fig. 45: Results of the ramp detection via the multi-parameter segmentation algorithm for $\gamma = 0.01, \varphi = 2$ and $P_{val} = 0.2$.

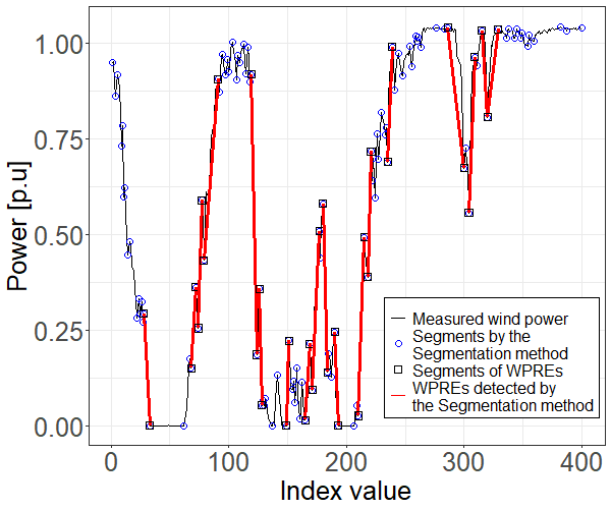


Fig. 46: Results of the ramp detection via the multi-parameter segmentation algorithm for $\gamma = 0.01, \varphi = 5$ and $P_{val} = 0.2$.

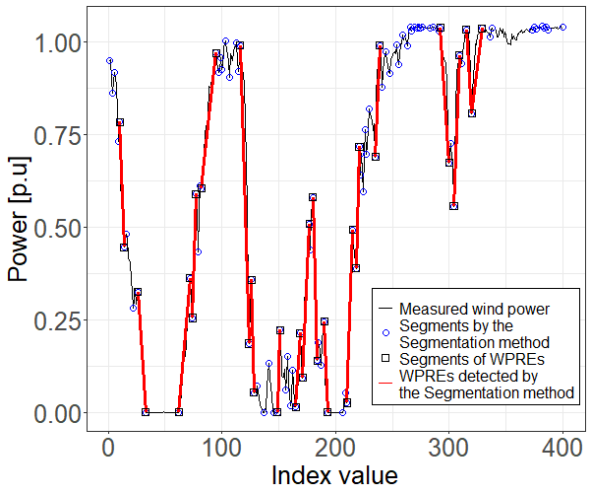


Fig. 47: Results of the ramp detection via the multi-parameter segmentation algorithm for $\gamma = 0.05, \varphi = 2$ and $P_{val} = 0.2$.

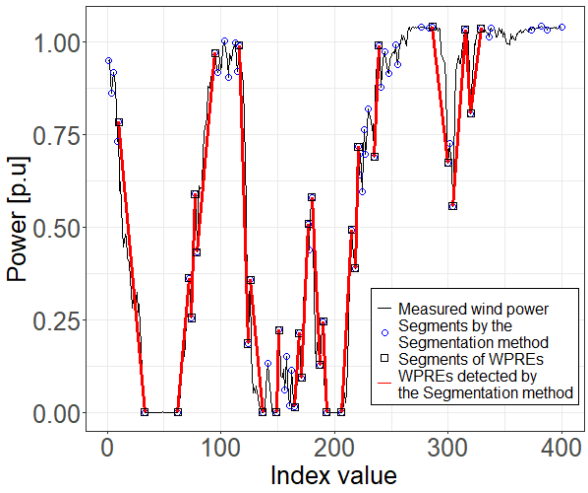


Fig. 48: Results of the ramp detection via the multi-parameter segmentation algorithm for $\gamma = 0.05, \varphi = 5$ and $P_{val} = 0.2$.

3.6.3 Statistical analysis of key ramp features

3.6.3.1 Ramp duration

Fig. 49 and Fig. 50 depict the Probability Density Function (PDF) of the ramp duration of the upward and downward ramps, respectively, detected using the multi-parameter segmentation algorithm for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$. It is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the multi-parameter segmentation algorithm will have a short duration rather than a long duration.

The probability density functions for the upward and downward ramps are approximately similar. The upward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$ have a mean ramp duration of 60.05 min, 60.07 min, 67.89 min, 70.98 min, 78.79 min and 104.31 min, respectively. For $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$, the maximum duration observed for the upward ramps is 270 min, 270 min, 310 min, 400 min, 330 min and 470 min, respectively. The minimum duration observed for the upward ramps detected using the multi-parameter segmentation algorithm for the various values of γ and φ is 10 minutes, respectively. Furthermore, for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$, respectively, 75% of the upward ramps have a duration less than or equal to 80 min, 80 min, 90 min, 90 min, 100 min and 140 min.

Furthermore, the downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$ have a mean ramp duration of 61 min, 61.46 min, 69.84 min, 73.31 min, 79.93 min and 105.6 min, respectively. Additionally, for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$, the maximum duration observed for the downward ramps is 270 min, 270 min, 330 min, 400 min, 330 min and 500 min, respectively. The minimum duration observed for the downward ramps detected using the multi-parameter segmentation algorithm for the

various values of γ and φ is 10 minutes, respectively. Furthermore, for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$, respectively, 75% of the downward ramps have a duration less than or equal to 80 min, 80 min, 90 min, 90 min, 100 min and 140 min.

Therefore, based on Fig. 49 and Fig. 50 as well as the information mentioned above, it is evident that the downward ramps detected by the multi-parameter segmentation algorithm generally have a slightly longer duration than the upward ramps. Additionally, if the size of γ decreases, the duration of the detected upward and downward ramps also decreases since the threshold for the permissible vertical fluctuation is more easily violated. For smaller values of γ, the effect of φ on the ramp duration is not very evident. However, for larger values of γ it is evident that if φ increases, the duration of the ramps also increases. Bearing in mind, the segment must remain within γ for a duration longer than φ to be reclassified to horizontal segments. Therefore, for a larger value of φ, this condition is not easily met resulting in the fluctuating segments forming part of the ramp occurrence, instead of being reclassified to horizontal segments, which increases the ramp duration.

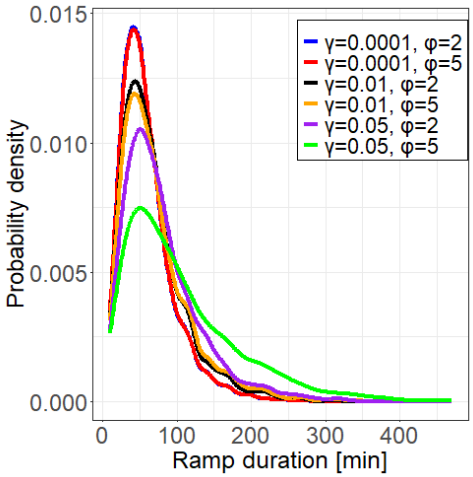


Fig. 49: Probability density functions of the ramp duration of upward ramps detected by the multi-parameter segmentation algorithm for various values of γ and φ.

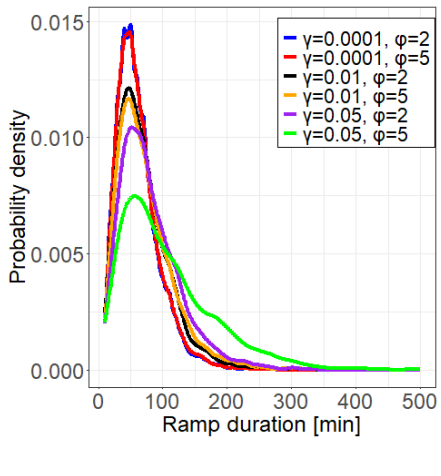


Fig. 50: Probability density functions of the ramp duration of downward ramps detected by the multi-parameter segmentation algorithm for various values of γ and φ.

3.6.3.2 Ramp magnitude

Fig. 51 and Fig. 52 depict the Probability Density Function (PDF) of the ramp magnitude of the upward and downward ramps, respectively, detected using the multi-parameter segmentation algorithm for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$. It is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the multi-parameter segmentation algorithm will have a small magnitude rather than a large magnitude.

The probability density functions for the upward and downward ramps are approximately similar. The upward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$ have a mean ramp magnitude of 0.3847 p.u., 0. 3847 p.u., 0.3922 p.u., 0. 3922 p.u., 0.4107 p.u. and 0.4297 p.u., respectively. The maximum magnitude observed for the upward ramps detected using the multi-

parameter segmentation algorithm for the various values of γ and φ is 1.0433 p.u., respectively. The minimum magnitude observed for the upward ramps detected using the multi-parameter segmentation algorithm for the various values of γ and φ is 0.20 p.u., respectively. Furthermore, for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$, respectively, 75% of the upward ramps have a magnitude less than or equal to 0.4640 p.u., 0.4640 p.u., 0.4775 p.u., 0.4772 p.u., 0.5066 p.u. and 0.5367 p.u., respectively.

Furthermore, the downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$ have a mean ramp magnitude of 0.3772 p.u., 0. 3772 p.u., 0.3834 p.u., 0.3835 p.u., 0.4 p.u. and 0.4184 p.u., respectively. The maximum magnitude observed for the downward ramps detected using the multi-parameter segmentation algorithm for the various values of γ and φ is 1.0219 p.u., 1.0219 p.u., 1.0394 p.u., 1.0395 p.u., 1.0394 p.u. and 1.040 p.u., respectively. The minimum magnitude observed for the downward ramps detected using the multi-parameter segmentation algorithm for the various values of γ and φ is 0.20 p.u., respectively. Furthermore, for $\gamma = 0.0001, \varphi = 2$; $\gamma = 0.0001, \varphi = 5$; $\gamma = 0.01, \varphi = 2$; $\gamma = 0.01, \varphi = 5$; $\gamma = 0.05, \varphi = 2$ and $\gamma = 0.05, \varphi = 5$, respectively, 75% of the downward ramps have a magnitude less than or equal to 0.4510 p.u., 0.4510 p.u., 0.4611 p.u., 0.4614 p.u., 0.4873 p.u. and 0.5130 p.u.

Based on Fig. 51 and Fig. 52 as well as the information mentioned above, it is evident that the downward ramps detected by the multi-parameter segmentation algorithm generally have a slightly smaller magnitude than the upward ramps detected by the multi-parameter segmentation algorithm. If the size of γ decreases, the magnitude of the detected upward and downward ramps also decreases since the threshold for the permissible vertical fluctuation is more easily violated. For smaller values of γ, the effect of φ on the ramp magnitude is not very evident. However, for larger values of γ it is evident that if φ increases, the magnitude of the ramps also increases. Bearing in mind, the segment must remain within γ for a duration longer than φ to be reclassified to horizontal segments. For a larger value of φ, this condition is not easily met resulting in the fluctuating segments forming part of the ramp occurrence, instead of being reclassified to horizontal segments, which may lead to ramps of increased magnitude.

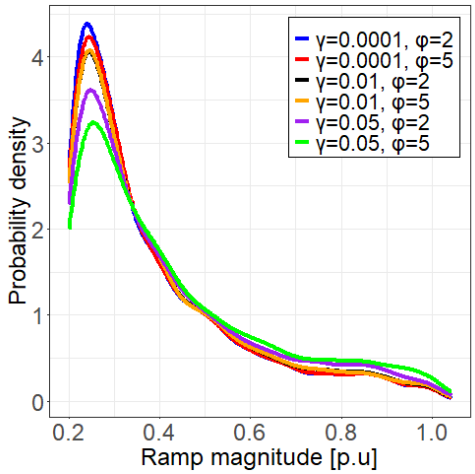


Fig. 51: Probability density functions of the ramp magnitude of upward ramps detected by the multi-parameter segmentation algorithm for various values of γ and φ.

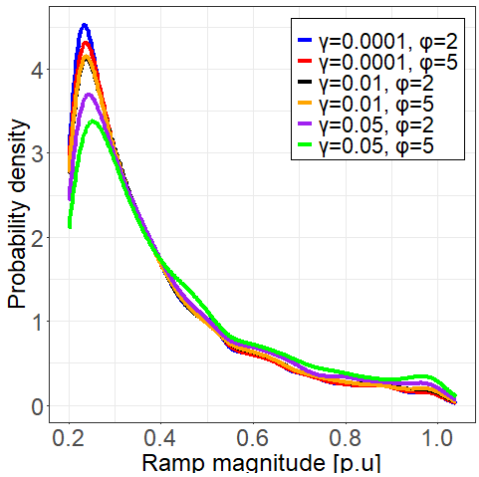


Fig. 52: Probability density functions of the ramp magnitude of downward ramps detected by the multi-parameter segmentation algorithm for various values of γ and φ.

3.7 A multi-parameter segmentation algorithm for wind power ramp detection with particle swarm optimisation

In this section, an original algorithm is proposed to decompose a temporal signal into upward and downward ramping segments. The algorithm is adapted from the multi-parameter segmentation algorithm proposed in section 3.6. Specifically, the application of the post-processing algorithm to discard horizontal segments is replaced by the application of particle swarm optimisation to identify optimal ramps. A detailed description of the complete algorithm is included below.

The initial power profile, P, is given by the set

$$P = \{p(t_i), \quad 0 \leq t_i \leq t_N\}. \tag{105}$$

The proposed algorithm, firstly, adopts an iterative evaluation of the raw signal in order to determine the underlying ramp features. This is accomplished by evaluating the signal trend using (97) and (98) simultaneously. A more detailed description of this procedure is included in section 3.6. Consequently, the power profile for a ramp segment can be expressed by the relationship

$$P_s(t_i, t_j) = \left\{ \begin{matrix} p_s(t_m), t_i \leq t_m \leq t_j, \\ (R_-(t_i, t_j) = 1) \vee (R_+(t_i, t_j) = 1) \end{matrix} \right\}. \tag{106}$$

Fig. 53 and Fig. 54 show the ramp segments that were extracted via the application of the proposed segmentation rules for $\gamma = 0.01$ and 0.05, respectively.

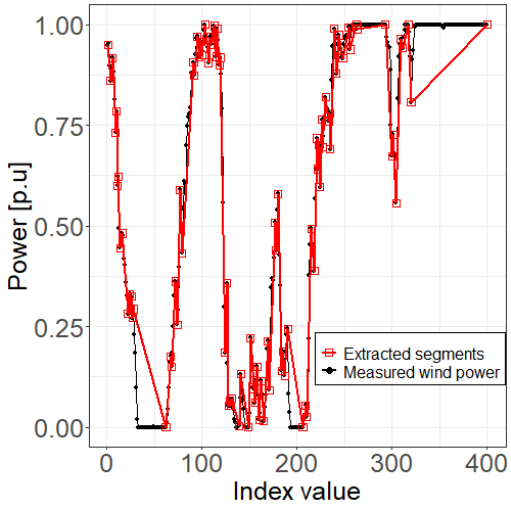


Fig. 53: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.01$.

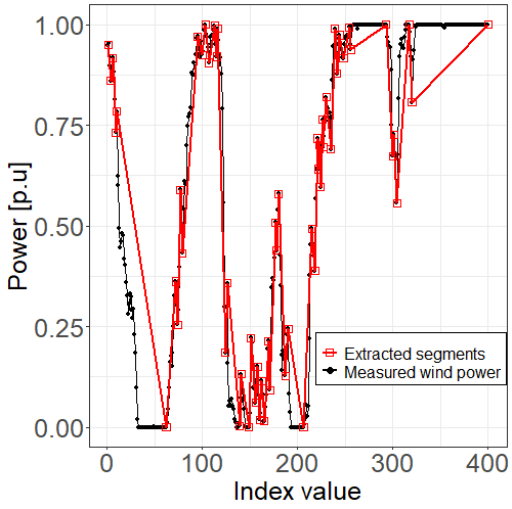


Fig. 54: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.05$.

It is recognised that three different signal trends are present within the ramp segments, namely within a ramp segment the signal can change rapidly, gradually or remain constant. A ramp, however, is defined as a large increase or decrease in wind power within a short period of time, i.e. a rapid change in wind power. The application of the particle swarm optimisation is, therefore, proposed to determine the start- and end-point of the optimal ramp, i.e. rapid changes, contained in P_s and correspondingly distinguish the optimal ramp contained in P_s from insignificant ramps.

The problem consists of finding the sub-interval within P_s that altogether minimises the ramp duration and maximises the ramp magnitude. Accordingly, the scoring functions $M(t_k, t_d)$ and $D(t_k, t_d)$ is used to assign a score to every possible sub-interval within P_s based on the magnitude and duration of the corresponding ramp, respectively, according to the relationships

$$M(t_k, t_d) = \frac{p_s(t_d) - p_s(t_k)}{p_s(t_j) - p_s(t_i)}, \qquad t_i \leq t_k < t_d \leq t_j \tag{107}$$

and

$$D(t_k, t_d) = \frac{t_d - t_k}{t_i - t_i}, \qquad t_i \leq t_k < t_d \leq t_j\,, \tag{108}$$

where the sub-interval is defined by $\mathrm{t_k}$ and $\mathrm{t_d}$. Given the scoring functions $\mathrm{M(t_k, t_d)}$ and $\mathrm{D(t_k, t_d)}$, the start- and end-point of the optimal ramp contained within $P_s(t_i, t_j)$ is recovered by minimising the objective function $J(\mathrm{t_i, t_j})$ given by the formulation

$$J(t_i, t_j) = \min_{t_i \leq t_k < t_d \leq t_j} w_1 \left(\frac{1}{M(t_k, t_d)}\right) + w_2 D(t_k, t_d)\,, \tag{109}$$

where

$$w_1 + w_2 = 1\,. \tag{110}$$

The weights assigned to $\mathrm{M(t_k, t_d)}$ and $\mathrm{D(t_k, t_d)}$, is represented by $\mathrm{w_1}$ and $\mathrm{w_2}$, respectively. The sub-interval defined by $\mathrm{t_k}$ and $\mathrm{t_d}$ that minimises $J(t_i, t_j)$ demarcates the optimal ramp within $P_s(t_i, t_j)$. The power profile for the optimal ramp contained in $P_s(t_i, t_j)$ can be expressed by the relationship

$$P_o(t_k, t_d) = \begin{Bmatrix} p_s(t_m), t_i \leq t_k \leq t_m \leq t_d \leq t_j, \\ J^*(t_i, t_j) \end{Bmatrix}, \tag{111}$$

where $\mathrm{J^*(t_i, t_j)}$ denotes the solution to (109).

When investigating the updated ramp segments, it was discovered that the correct start- and end-points were not always found for the remainder-ramps, i.e. those ramps outside the boundaries of the recovered optimal ramps $\mathrm{P_o(t_k, t_d)}$ but still within the initial ramp segments $P_s(t_i, t_j)$. It is, therefore, necessary to also apply the particle swarm optimisation to these remainder-ramps in order to identify the optimal ramps contained in these remainder-ramps. The particle swarm optimisation is, however, only applied to the remainder-ramps that exceed a user-specified magnitude threshold, since the main objective of ramp extraction is to identify significant ramps within the temporal wind power profile, which are defined in this study as ramps that exceed some magnitude threshold. Extracting the correct start- and end-points of insignificant ramps is, therefore, not important and unnecessarily prolongs the run time of the algorithm.

This process of extracting optimal ramps from remainder-ramps that exceed a magnitude threshold is performed recursively for completeness. This means that the particle swarm optimisation is applied to all remainder-ramps that exceed the magnitude threshold, and then to sub-remainder-ramps, which result from demarcating optimal ramps within the remainder-ramps, that exceed the magnitude threshold and so on. Evidently, it was only necessary to apply the particle swarm optimisation to the remainder-ramps,

and it was never required to apply the particle swarm optimisation to sub-remainder-ramps or deeper levels.

The result of the complete algorithm for different values of γ, as well as w_1 and w_2, is shown in Fig. 55 to Fig. 62. It is recognised that as the size of w_1 increases, correspondingly w_2 decreases. More importance is, therefore, placed on detecting a ramp that increases the ramp magnitude, while less importance is placed on decreasing the ramp duration. Accordingly, the magnitude and duration of the extracted ramp segments increases as w_1 increases, which in turn decreases the number of ramp segments used to linearly approximate the signal.

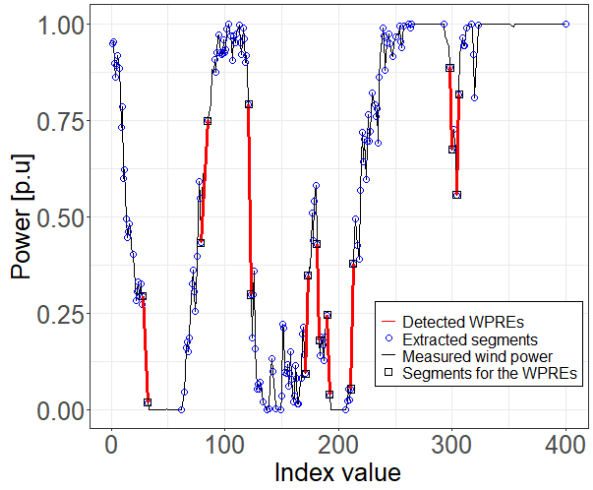


Fig. 55: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.01$ combined with the application of the particle swarm optimisation for $w_1 = 0.25$ and $w_2 = 0.75$.

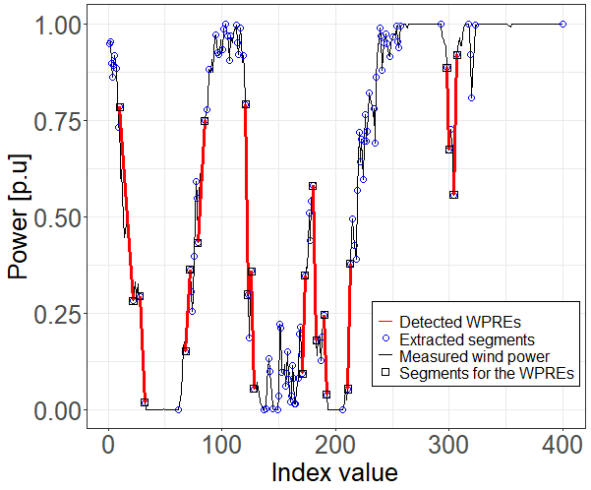


Fig. 56: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.05$ combined with the application of the particle swarm optimisation for $w_1 = 0.25$ and $w_2 = 0.75$.

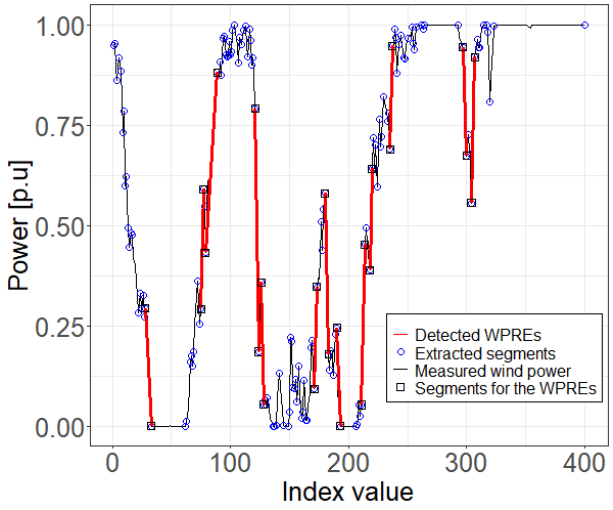


Fig. 57: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.01$ combined with the application of the particle swarm optimisation for $w_1 = 0.5$ and $w_2 = 0.5$.

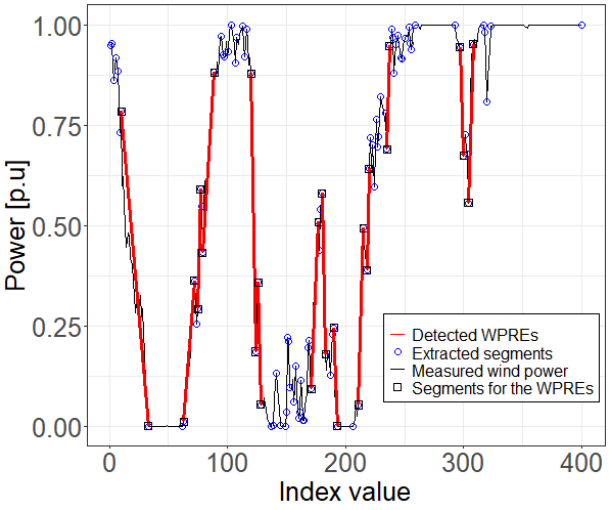


Fig. 58: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.05$ combined with the application of the particle swarm optimisation for $w_1 = 0.5$ and $w_2 = 0.5$.

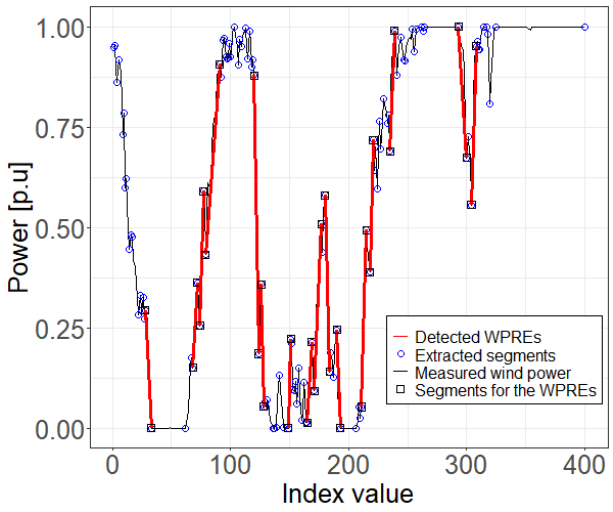


Fig. 59: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.01$ combined with the application of the particle swarm optimisation for $w_1 = 0.75$ and $w_2 = 0.25$.

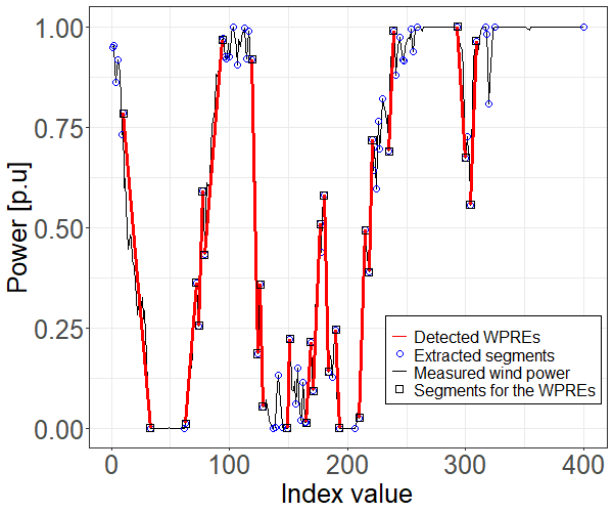


Fig. 60: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.05$ combined with the application of the particle swarm optimisation for $w_1 = 0.75$ and $w_2 = 0.25$.

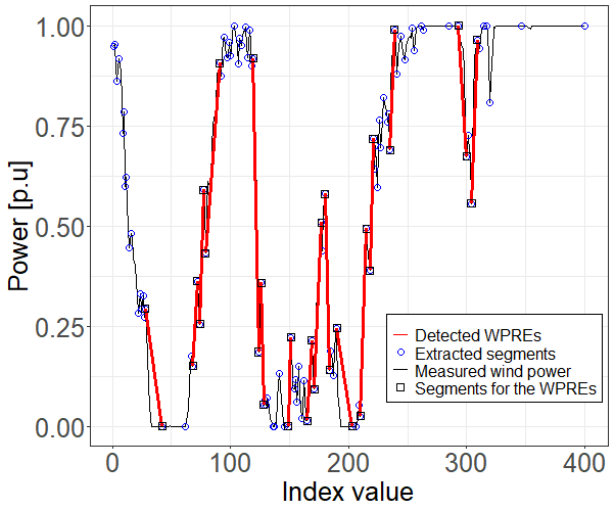


Fig. 61: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.01$ combined with the application of the particle swarm optimisation for $w_1 = 1$ and $w_2 = 0$.

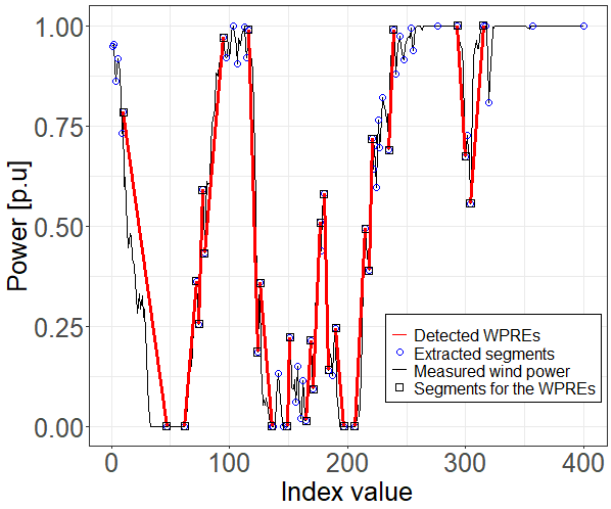


Fig. 62: The ramps extracted via the application of the upward and downward segmentation rules for $\gamma = 0.05$ combined with the application of the particle swarm optimisation for $w_1 = 1$ and $w_2 = 0$.

The proposed algorithm successfully identifies optimal ramps within the signal. The ramp model employed in this section is consistent with the informal definition of a ramp event, i.e. a large increase in wind power within a short period of time. The proposed algorithm also allows for the consideration of weights that determine the contribution and relative importance of the ramp duration and ramp magnitude in identifying the optimal ramps.

3.8 Regression-based segmentation algorithms

3.8.1 Overview

In this section the application of a simple linear regression model is proposed to extract ramp events from a temporal wind power profile. Ramps are regarded as intervals in which the wind power is considered to follow a single trend. A simple linear regression model incorporates only one regressor variable and describes the relationship between the response variable and the regressor variable using a straight-line. The simple linear regression model is, therefore, suitable to represent a ramp event.

The initial power profile, P, is given by

$$P = \{p(t_i), \quad 0 \leq t_i \leq t_N\}, \tag{112}$$

where N denotes the number of time instants in the profile.

With a simple linear regression model, it is possible to extract ramps from a signal in a piecewise linear fashion. When exploring the relationship between the response variable, $p(t_i)$, and the regressor variable, t_i, it is evident that different linear relationships, i.e. trends, are present for different ranges of t_i. The different linear relationships correspond to the ramps of the signal. It is, therefore, necessary to determine the break points in the signal, i.e. the points where the linear relationships change, so to demarcate the ramp intervals. To accomplish this, a simple linear regression model is fit to a window of data points increasing iteratively in length. The simple linear regression model is estimated using the least squares estimation method. At each iteration, key goodness-of-fit measures are evaluated for the simple linear regression model. This allows for the consideration of threshold parameters for these key goodness-of-fit measures, defining the objectives for an adequate model and so affecting the identified ramps. A ramp is found when the threshold for the applicable key goodness-of-fit measure of the simple linear regression model is violated. The end point of a ramp in turn becomes the starting point of the following segment. This process continues until all samples in the wind power time series are evaluated. This iterative process ensures that the linear regression model representing a ramp interval satisfies the user-specified objectives for an adequate model based on satisfying the requirements for key goodness-of-fit measures.

The key goodness-of-fit measures of the simple linear regression model for which a threshold is considered, include the following:

- *Coefficient of determination*: The coefficient of determination, R^2, is the fraction of the total variability in the response variable that is explained by the simple linear regression model. The value of R^2 ranges from 0 to 1, with 1 indicating a perfect fit and 0 indicating that the regression line is horizontal and correspondingly there is no linear relationship between the response variable and the regressor variable.
- *Standard error of regression*: The standard error of regression measures the typical difference between the observed value of the response variable and its corresponding fitted value, i.e. the typical value of the residual. The units of the standard error of regression correspond to that of the response variable. The standard error of regression should be as low as possible, since this means that the difference between the observed values of the response variable and the fitted values are typically small.
- *Maximum residual*: The standard error of regression has a drawback associated with its use. By considering the average value of the residual, valuable information about the variability of the

residuals and, specifically, the extreme residual values are lost. A simplified approach to address the drawback associated with the standard error of regression involves thresholding the maximum difference between the observed value of the response variable and the corresponding fitted value, i.e. the maximum residual. The maximum residual should be as low as possible, so to ensure all residual values are minimised.

- *The slope of the regression line*: The slope of the regression line, i.e. the gradient, defines the amount the response variable changes by when the regressor variable increases by one unit, thereby, signifying the linear relationship between the response variable and the regressor variable. Adopting an iterative evaluation of the signal trend via the slope of the estimated regression lines may suggest a good estimate for the break points in the signal. Specifically, when the slope of the estimated regression line changes significantly, it may be an indication of a break point in the signal.

The following sections present the implementation of the ramp detection algorithms based on the application of the simple linear regression model and the consideration of a threshold parameter for key goodness-of-fit measures of the simple linear regression model, referred to as regression-based segmentation algorithms.

3.8.2 Regression-based segmentation algorithm considering a threshold for the coefficient of determination

The initial power profile, P, is given by (112).

The direction, duration, and magnitude of each upward and downward ramp occurrence is initially unknown. The proposed algorithm is implemented by iteratively extending the end point defined by j from a starting point defined by i, and estimating the regression line for the segment defined by t_i and t_j. $R_{\mathrm{R2}}(i,j)$ is calculated for the segment defined by t_i and t_j, according to the relationship

$$R_{R2}(t_i,t_j) = \mathbf{1}_{\{[R^2 \geq \alpha\}}\,. \qquad (113)$$

This rule determines whether the coefficient of determination R^2 of the regression line fitted to the segment defined by t_i and t_j exceeds the threshold denoted by α. $\mathrm{R}_{\mathrm{R2}}(\mathrm{t_i},\mathrm{t_j})$ returns a 1 if the condition is met and 0 otherwise. The end of a ramp is defined as the time instant t_j before $R_{\mathrm{R2}}(t_i,t_j)$ equals 0. The end point in turn becomes the starting point of the following segment. This process continues until all samples in the wind power time series are evaluated. The signal is normalised before the algorithm is executed. The power profile for a ramp segment can be expressed by the relationship

$$P_s(t_i,t_j) = \begin{Bmatrix} p_s(t_m), t_i \leq t_m \leq t_j, \\ (R_{R2}(t_i,t_k) = 1, t_i < t_k \leq t_j) \end{Bmatrix}. \qquad (114)$$

After the algorithm is applied to the wind power time series to extract the linear ramps, a user specified definition of a significant ramp is used to identify the wind power ramp events present in the signal.

Fig. 63 to Fig. 66 show the segments and the corresponding significant ramps extracted via the application of the simple linear regression model, by thresholding the coefficient of determination by $\alpha = 0.8, 0.85, 0.9$ and 0.95, respectively. Higher values for the coefficient of determination represent smaller residual values as the regression model must explain a larger percentage of the total variability in the response variable. The magnitude and duration of the segments used to linearly approximate the signal therefore decreases as the threshold parameter for the coefficient of determination increases, since the

threshold is more easily violated. This in turn increases the number of segments and ensures a more accurate approximation of the signal. Accordingly, the threshold parameter for the coefficient of determination introduces a trade-off between accuracy and the number of segments.

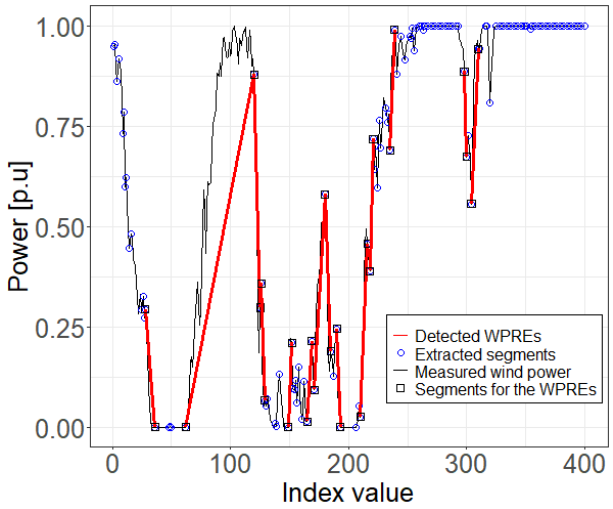


Fig. 63: The ramps detected via the application of the simple linear regression model by thresholding the coefficient of determination by $\alpha = 0.8$.

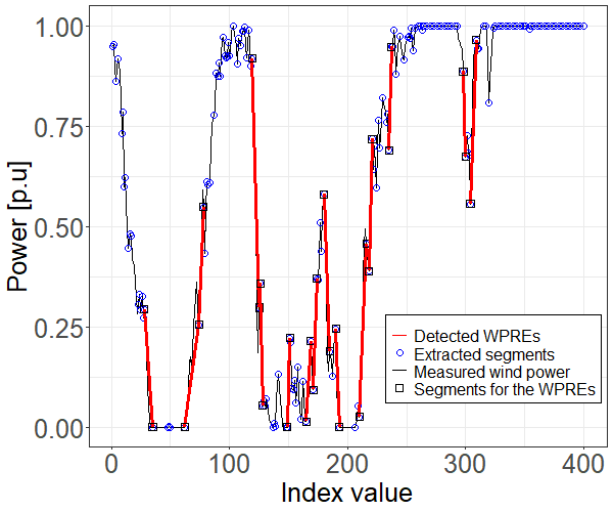


Fig. 64: The ramps detected via the application of the simple linear regression model by thresholding the coefficient of determination by $\alpha = 0.85$.

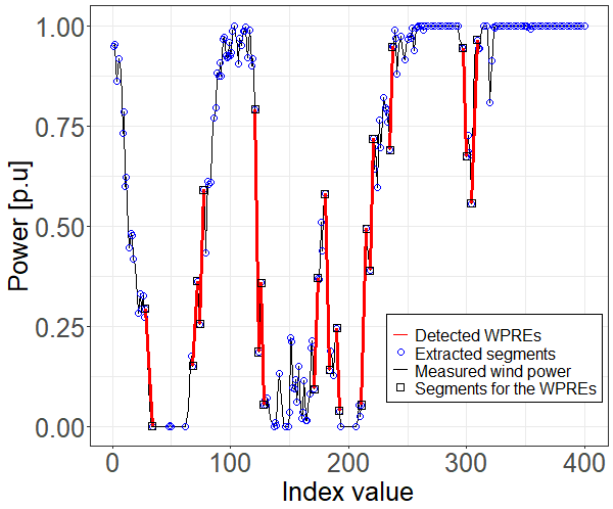


Fig. 65: The ramps detected via the application of the simple linear regression model by thresholding the coefficient of determination by $\alpha = 0.9$.

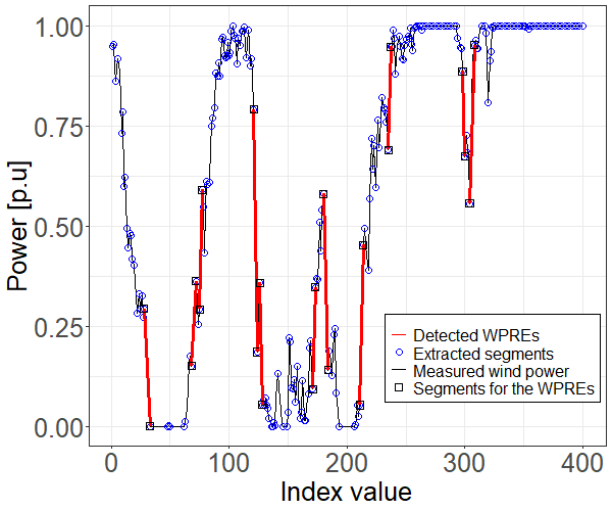


Fig. 66: The ramps detected via the application of the simple linear regression model by thresholding the coefficient of determination by $\alpha = 0.95$.

Table 8 depicts the run-time of the ramp detection algorithm based on the application of the simple linear regression model, by thresholding the coefficient of determination by $\alpha = 0.8, 0.85, 0.9$ and 0.95, respectively. The run-time of the algorithm increases as the tunable parameter for the coefficient of determination increases.

Table 8: Run-time of the ramp detection algorithm based on the application of the simple linear regression model by thresholding the coefficient of determination by $\alpha = 0.8, 0.85, 0.9$ and 0.95, respectively.

Tolerance for the coefficient of determination	Run-time [min]
0.8	8.1950
0.85	10.2694
0.9	15.0272
0.95	23.7663

Several limitations are associated with thresholding the coefficient of determination. The power generation profiles associated with wind power sources exhibit a relatively high degree of variability and uncertainty, i.e. wind power has a high degree of unexplainable variation. The coefficient of determination of the simple linear regression models estimating the wind power ramps are, therefore, expected to be lower, and specifying a value that is too high for the threshold parameter of the coefficient of determination may result in overfitting. Additionally, the degree of variability in the wind power also varies across different ranges of the regressor variable. It is, therefore, not obvious what is considered as a suitable threshold for the coefficient of determination.

It is possible to minimise the number of segments that linearly approximates the signal by optimising the segments extracted via the proposed algorithm using the dynamic recursion technique adopted from Sevlian and Rajagopal [21].

3.8.3 Regression-based segmentation algorithm considering a threshold for the standard error

The initial power profile, P, is given in (112).

The direction, duration, and magnitude of each upward and downward ramp occurrence is initially unknown. The proposed algorithm is implemented by iteratively extending the end point defined by j from a starting point defined by i, and estimating the regression line for the segment defined by t_i and t_j. $R_{\mathrm{SE}}(i,j)$ is calculated for the segment defined by t_i and t_j, according to the relationship

$$R_{SE}(t_i, t_j) = \mathbf{1}_{\{[\hat{\sigma} \leq \alpha\}}\,. \tag{115}$$

This rule determines whether $\hat{\sigma}$, representing the standard error of regression of the regression model fitted to the segment defined by t_i and t_j, is less than or equal to the threshold denoted by α. $\mathrm{R_{SE}(t_i, t_j)}$ returns a 1 if the condition is met and 0 otherwise. The end of a ramp is defined as the time instant t_j before $\mathrm{R_{SE}(t_i, t_j)}$ equals 0. The end point in turn becomes the starting point of the following segment. This process continues until all samples in the wind power time series are evaluated. The signal is normalised before the algorithm is executed. The power profile for a ramp segment can be expressed by the relationship

$$P_s(t_i, t_j) = \begin{Bmatrix} p_s(t_m), t_i \leq t_m \leq t_j, \\ (R_{SE}(t_i, t_k) = 1, t_i < t_k \leq t_j) \end{Bmatrix}. \tag{116}$$

After the algorithm is applied to the wind power time series to extract the linear ramps, a user specified definition of a significant ramp is used to identify the wind power ramp events present in the signal.

Fig. 67 to Fig. 70 show the segments and significant ramps extracted via the application of the simple linear regression model, by thresholding the standard error of regression by $\alpha = 0.005, 0.01, 0.025$ and 0.05, respectively. The standard error of regression measures the typical difference between the observed value of the response variable and its corresponding fitted value. Lower values for the standard error of regression, therefore, represent smaller residual values. The magnitude and duration of the segments used to linearly approximate the signal decreases as the threshold parameter for the standard error of regression decreases, since the threshold is more easily violated. This in turn increases the number of segments and ensures a more accurate approximation of the signal. Accordingly, the threshold parameter for the standard error of regression introduces a trade-off between accuracy and the number of segments.

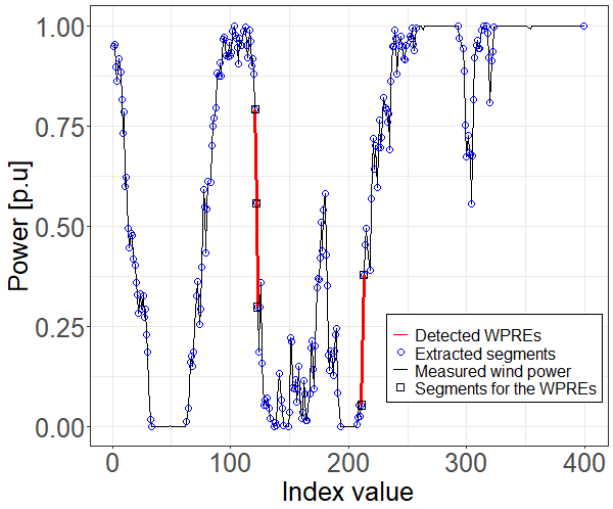


Fig. 67: The ramps detected via the application of the simple linear regression model by thresholding the standard error of regression by $\alpha = 0.005$.

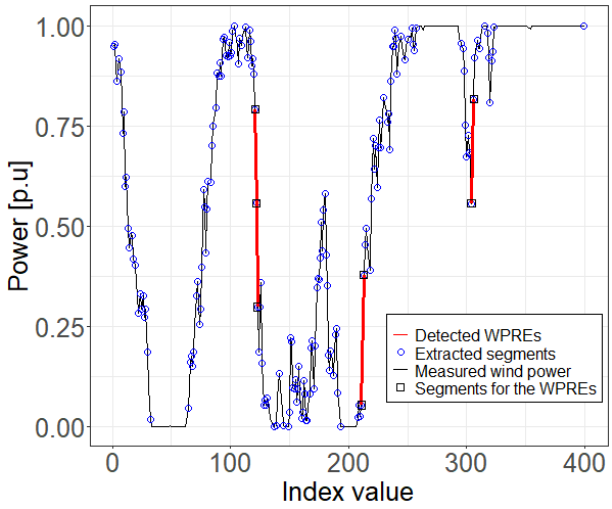


Fig. 68: The ramps detected via the application of the simple linear regression mode, by thresholding the standard error of regression by $\alpha = 0.01$.

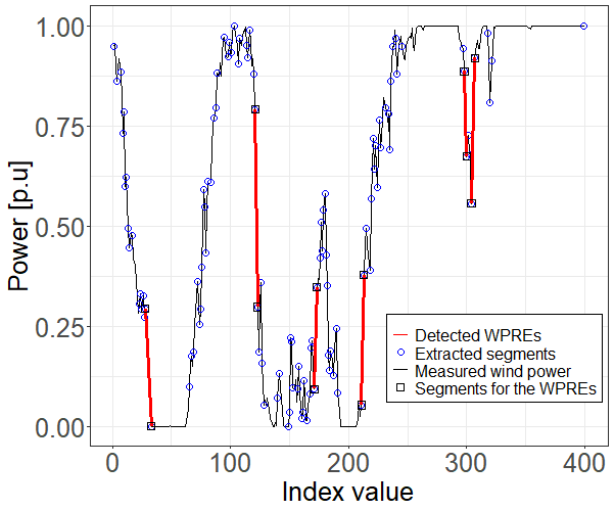


Fig. 69: The ramps detected via the application of the simple linear regression model by thresholding the standard error of regression by $\alpha = 0.025$.

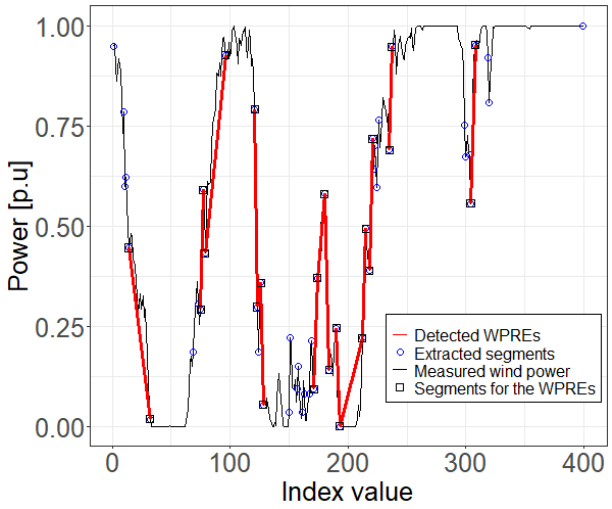


Fig. 70: The ramps detected via the application of the simple linear regression model by thresholding the standard error of regression by $\alpha = 0.05$.

Table 9 depicts the run-time of the ramp detection algorithm based on the application of the simple linear regression model, by thresholding the standard error of regression by $\alpha = 0.005, 0.01, 0.025$ and 0.05, respectively. The run-time of the algorithm increases as the tunable parameter for the standard error of regression decreases.

Table 9: Run-time of the ramp detection algorithm based on the application of the simple linear regression model, by thresholding the standard error of regression by $\alpha = 0.005, 0.01, 0.025$ and 0.05, respectively.

Tolerance for the standard error	Run-time [min]
0.005	33.0016
0.01	23.1311
0.25	13.9526
0.05	9.0468

The advantage of thresholding the regression-based segmentation algorithm with the standard error of regression is that the precision of the model is measured in terms of the units of the response variable, instead of a percentage as for the regression-based segmentation algorithm considering a threshold for the coefficient of determination, thereby, making the model more intuitive.

It is possible to minimise the number of segments that linearly approximates the signal by optimising the segments extracted via the proposed algorithm using the dynamic recursion technique adopted from Sevlian and Rajagopal [21].

3.8.4 Regression-based segmentation algorithm considering a threshold for the maximum residual

The initial power profile, P, is given in (112).

The direction, duration, and magnitude of each upward and downward ramp occurrence is initially unknown. The proposed algorithm is implemented by iteratively extending the end point defined by j from a starting point defined by i, and estimating the regression line for the segment defined by t_i and t_j. $R_{\max}(i,j)$ is calculated for the segment defined by t_i and t_j, according to the relationship

$$R_{max}(t_i, t_j) = \mathbf{1}_{\left\{\max\limits_{i \le m \le j}(e_m) \le \alpha\right\}}, \tag{117}$$

with e_m representing the m^{th} residual for the segment defined by t_i and t_j. This rule determines whether the maximum residual for the segment defined by t_i and t_j is less than or equal to the threshold denoted by α. $\mathrm{R_{max}(t_i, t_j)}$ returns a 1 if the condition is met and 0 otherwise. The end of a ramp is defined as the time instant t_j before $R_{\max}(i,j)$ equals 0. The end point in turn becomes the starting point of the following segment. This process continues until all samples in the wind power time series are evaluated. The signal is normalised before the algorithm is executed. The power profile for a ramp segment can be expressed by the relationship

$$P_s(t_i, t_j) = \begin{Bmatrix} p_s(t_m), t_i \le t_m \le t_j, \\ (R_{max}(t_i, t_k) = 1, t_i < t_k \le t_j) \end{Bmatrix}. \tag{118}$$

After the algorithm is applied to the wind power time series to extract the linear ramps, a user specified definition of a significant ramp is used to identify the wind power ramp events present in the signal.

Fig. 71 to Fig. 74 show the segments and significant ramps extracted via the application of the simple linear regression model, by thresholding the maximum residual value by $\alpha = 0.005, 0.01, 0.025$ and 0.05, respectively. The magnitude and duration of the segments used to linearly approximate the signal decreases as the threshold parameter for the maximum residual value decreases, since the threshold is more easily violated. This in turn increases the number of segments and ensures a more accurate approximation of the signal. Accordingly, the threshold parameter for the maximum residual value introduces a trade-off between accuracy and the number of segments.

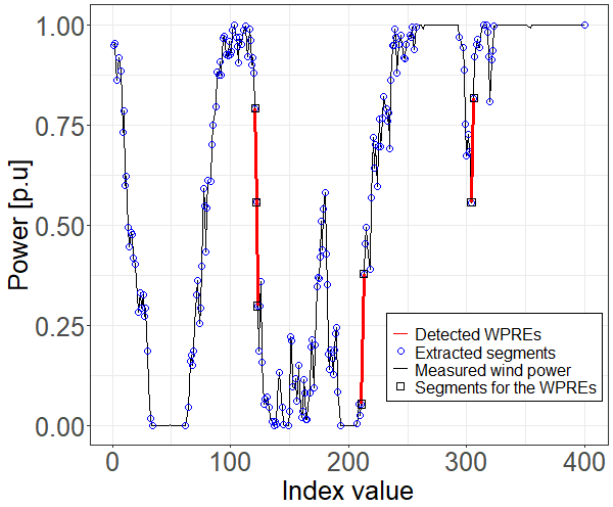


Fig. 71: The ramps detected via the application of the simple linear regression model by thresholding the maximum residual value by $\alpha = 0.005$.

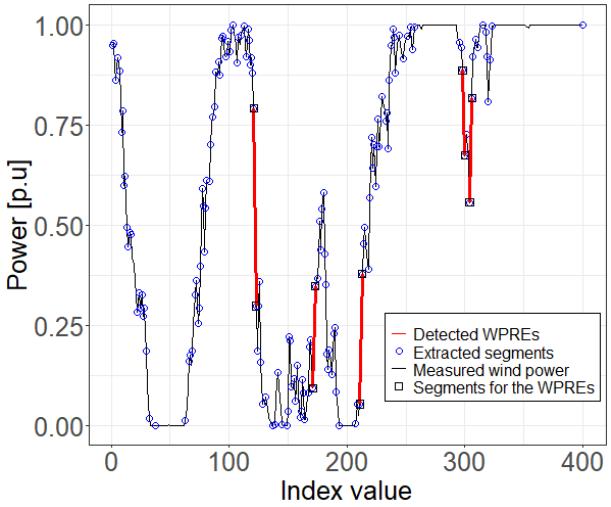


Fig. 72: The ramps detected via the application of the simple linear regression model by thresholding the maximum residual value by $\alpha = 0.01$.

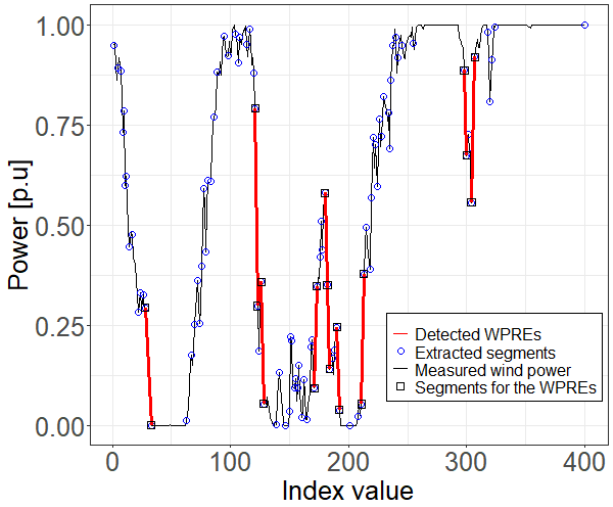


Fig. 73: The ramps detected via the application of the simple linear regression model by thresholding the maximum residual value by $\alpha = 0.025$.

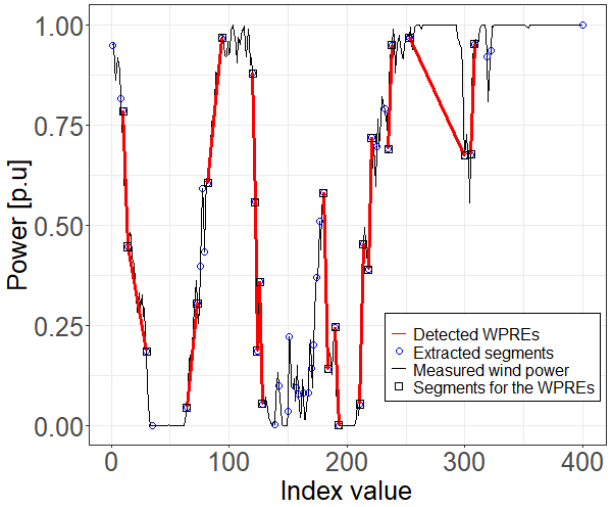


Fig. 74: The ramps detected via the application of the simple linear regression model by thresholding the maximum residual value by $\alpha = 0.05$.

Table 10 depicts the run-time of the ramp detection algorithm based on the application of the simple linear regression model, by thresholding the maximum residual value by $\alpha = 0.005, 0.01, 0.025$ and 0.05,

respectively. The run-time of the algorithm increases as the tunable parameter for the maximum residual value decreases.

Table 10: Run-time of the ramp detection algorithm based on the application of the simple linear regression model, by thresholding the maximum residual value by $\alpha = 0.005, 0.01, 0.025 \; and \; 0.05$ *.*

Tolerance for the maximum residual	Run-time [min]
0.005	17.8921
0.01	10.8661
0.25	5.5596
0.05	4.3128

The advantage of thresholding the regression-based segmentation algorithm with the maximum residual value is that the precision of the model is measured in terms of the units of the response variable, similar to thresholding the standard error of regression. The regression based segmentation algorithm thresholding the maximum residual, however, ensures that all the residual values of the simple linear regression model representing a ramp is less than or equal to the chosen threshold, whereas thresholding the standard error of regression ensures that the average value of the residuals are less than or equal to the chosen threshold.

It is possible to minimise the number of segments that linearly approximates the signal by optimising the segments extracted via the proposed algorithm using the dynamic recursion technique adopted from Sevlian and Rajagopal [21].

3.8.5 Regression-based segmentation algorithm considering a threshold for the slope

An investigation was performed by iteratively extending the end point defined by j from a starting point defined by i, and estimating the regression line for the segment defined by t_i and t_j. Characteristics of the fitted regression line for the segment defined by t_i and t_j, i.e. the coefficients and key goodness-of-fit measures of the fitted regression lines, were investigated to determine whether it may suggest optimal break points for the signal. It was found that break points in the signal corresponded to the flattening of the slope of the fitted regression lines, i.e. a decrease in the slope of the fitted regression line. This finding led to the development of the regression-based segmentation algorithm considering a threshold for the slope of the fitted regression line.

The initial power profile, P, is given in (112).

The direction, duration, and magnitude of each upward and downward ramp occurrence is initially unknown. The proposed algorithm is implemented by iteratively extending the end point defined by j from a starting point defined by i, and estimating the regression line for the segment defined by t_i and t_j. $\mathrm{R}_{+}(\mathrm{t_i}, \mathrm{t_j})$, $\mathrm{R}_{-}(\mathrm{t_i}, \mathrm{t_j})$ and $\mathrm{R}_{\mathrm{H}}(\mathrm{t_i}, \mathrm{t_j})$, denoting the increasing-, decreasing- and horizontal-segmentation rule, respectively, are calculated for the segment defined by t_i and t_j, according to the relationships

$$R_{+}(t_i,t_j)=\begin{cases}\mathbf{1}\ if\ j=i+1\wedge\ \beta(t_i,t_j)>0\\ \mathbf{1}\ if\ \ j>i+1\ \wedge\ \beta(t_i,t_j)>0\ \ \wedge\\ (abs\left(\beta(t_i,t_j)\right)\geq\frac{abs(\beta(t_i,t_{j-1})\times(t_{j-1}-t_i)}{t_j-t_i})\\ \mathbf{0}\ otherwise\end{cases} \quad (119)$$

$$R_{-}(t_i,t_j)=\begin{cases}\mathbf{1}\ if\ j=i+1\ \wedge\ \ \beta(t_i,t_j)<0\\ \mathbf{1}\ if\ \ j>i+1\ \wedge\beta(t_i,t_j)<0\ \ \wedge\\ (abs\left(\beta(t_i,t_j)\right)\geq\frac{abs(\beta(t_i,t_{j-1})\times(t_{j-1}-t_i)}{t_j-t_i})\\ \mathbf{0}\ otherwise\end{cases} \quad (120)$$

and

$$R_H(t_i,t_j)=\ \mathbf{1}_{\{\beta(t_i,t_j)=0\}}, \quad (121)$$

where $\beta(t_i,t_j)$ denotes the slope of the regression line estimated for the segment defined by t_i and t_j. The end of a ramp is defined as the time instant t_j before the applicable segmentation rule, i.e. $R_{+}(t_i,t_j)$, $R_{-}(t_i,t_j)$ or $R_H(t_i,t_j)$, equals 0. The end-point in turn becomes the start-point of the following segment. This process continues until all samples in the wind power time series are evaluated. The power profile for a ramp segment can be expressed by the relationship

$$P_s(t_i,t_j)=\left\{\begin{matrix}p_s(t_m),t_i\leq t_m\leq t_j,\\ (R_{+}(t_i,t_j)=1)\vee(R_{-}(t_i,t_j)=1)\vee(R_H(t_i,t_j)=1)\end{matrix}\right\}. \quad (122)$$

Fig. 75 shows the segments and significant ramps extracted via the application of the simple linear regression model, by thresholding the slope of the fitted regression line.

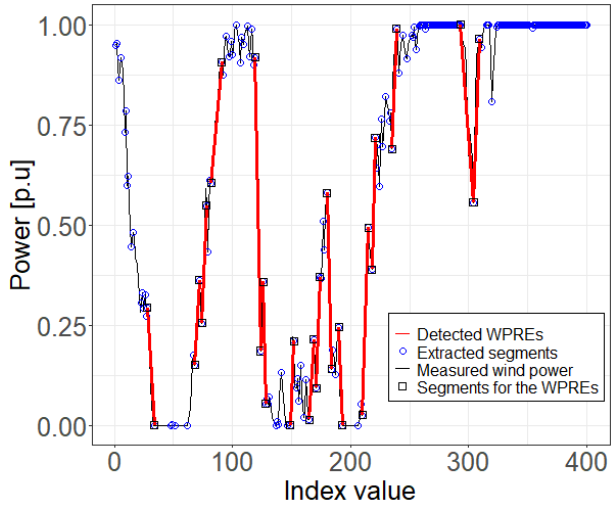


Fig. 75: The ramps detected via the application of the simple linear regression model by thresholding the slope of the fitted regression line.

The run-time for the proposed algorithm is 9.2178 minutes.

The method performs satisfactorily based on visual inspection. The drawback of the method is, however, that it does not consider a threshold parameter to allow ramp variations and define the sensitivity to ramp variations.

3.9 Comparison of detection algorithms

3.9.1 Overview

This section presents a comparison of the significant ramps extracted via the multi-parameter segmentation algorithm to the significant ramps extracted via the SDA, OpSDA and the L1-SW respectively, for optimal tunable parameter values and the selected ramps definition, based on various comparison methodologies to determine optimality. The comparison methodologies include visual inspection of the sets of ramps, a comparison of the run-times of the algorithms and the number of ramps, the implementation of a mathematical framework to determine equivalent ramps, the determination of the detection accuracy of the start- and end-points of the sets of ramps, as well as the comparison of the key ramp features.

The optimal tunable parameter values chosen for the SDA, OpSDA and L1-SW are consistent with values found in literature, and it is important to note that no attempt was made to solve the optimal tunable parameter values for these methods. Specifically, for the SDA and OpSDA an $\varepsilon = 0.009$ was chosen. For the L1-SW different parameter values were used, namely $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ and $\lambda = 0.5, \gamma = 1 \times 10^{-4}$. The optimal tunable parameter value for the multi-parameter segmentation algorithm was chosen as $\gamma = 0.1, \varphi = 5$ based on visual inspection. The parameters were chosen in order to produce similar significant ramps to those detected by the L1-SW and OpSDA. Furthermore, it is recognised that the comparison of the significant ramps is dependent on the choice of parameters, i.e. if different parameter values were chosen for each method, different significant ramps would have been detected and the results of the comparison would be different. This also reflects the lack of agreement on a formal ramp definition. Accordingly, the emphasis is on the comparison methodology, as well as the presented results, and not on solving the optimal tunable parameter values.

3.9.2 Ramp comparison based on visual inspection

Fig. 76 compares the significant ramps extracted via the multi-parameter segmentation algorithm for $\gamma = 0.1, \varphi = 5$ to the significant ramps extracted via the SDA for $\varepsilon = 0.009$ using the selected ramp definition. It is evident that more ramps are detected via the multi-parameter segmentation algorithm than the SDA. The ramps detected by the multi-parameter segmentation algorithm also have a greater magnitude and duration than the ramps detected via the SDA. The proposed model is superior to the SDA, especially with regards to the determination of start- and end-points of ramps. When comparing visually detected ramps to the ramps detected by both methods, the multi-parameter segmentation algorithm performs better. Any differences in the visually detected ramps and the ramps detected using the multi-parameter segmentation algorithm is due to the user-specified definition of the permissible vertical fluctuations, i.e. the size of γ and φ, as well as the user-specified definitions of the ramps.

Fig. 77 compares the significant ramps extracted via the multi-parameter segmentation algorithm for $\gamma = 0.1, \varphi = 5$ to the significant ramps extracted via the OpSDA for $\varepsilon = 0.009$, using the selected ramp definition. The significant ramps extracted by the multi-parameter segmentation algorithm and the

OpSDA are very similar, except for a few differences. The most left ramp was detected by the OpSDA but not detected by the multi-parameter segmentation algorithm. This is due to the user-specified definition of the size and length of the permissible fluctuations which can be present in the ramp, i.e. the size of γ and φ. For the same reason, the duration and magnitude of the significant ramps detected by the OpSDA are sometimes greater than that of the significant ramps detected by the multi-parameter segmentation algorithm. One ramp was also detected by the multi-parameter segmentation algorithm but incorrectly missed by the OpSDA. Furthermore, the multi-parameter segmentation algorithm performs better than the OpSDA with regards to the determination of start- and end-points of ramps, i.e. the detected ramps of the multi-parameter segmentation algorithm always end on a local minima or maxima, whereas the detected ramps of the OpSDA do not. The ramps detected by the OpSDA are also sometimes interjected by a horizontal segment which wrongly extends the length of the significant ramp. Conversely, the multi-parameter segmentation algorithm included a post-processing step to discard these horizontal segments from the significant ramps. Both sets of ramps correspond well to the visually detected ramps.

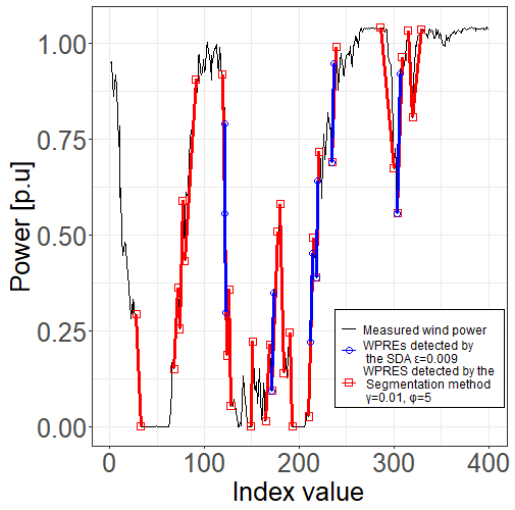


Fig. 76: A comparison of the significant ramps detected by the SDA for $\varepsilon = 0.009$ to the significant ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.1, \varphi = 5$.

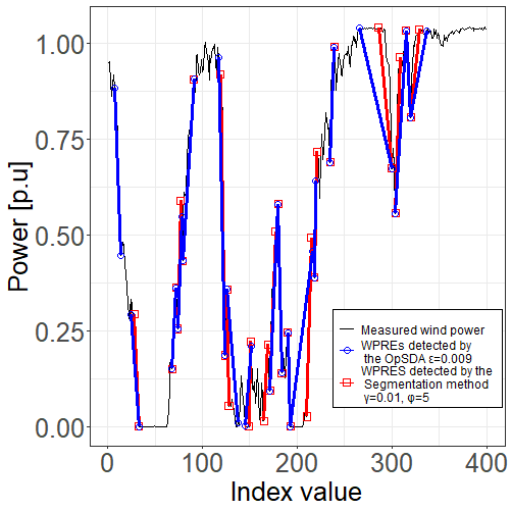


Fig. 77: A comparison of the significant ramps detected by the OpSDA for $\varepsilon = 0.009$ to the significant ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.1, \varphi = 5$.

Fig. 78 and Fig. 79 compare the significant ramps extracted via the multi-parameter segmentation algorithm for $\gamma = 0.1, \varphi = 5$ to the significant ramps extracted via the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ and $\lambda = 0.5, \gamma = 1 \times 10^{-4}$, respectively, using the selected ramp definition. As mentioned previously, the smaller values of λ and γ result in the detection of smaller ramps with shorter duration that also fits the signal more closely, when compared to using the L1-SW with greater λ values.

For the case of $\lambda = 0.02, \gamma = 5 \times 10^{-6}$, several of the significant ramps extracted by the multi-parameter segmentation algorithm and the L1-SW are approximately similar. However, several ramps were detected by the multi-parameter segmentation algorithm but not by the L1-SW. Due to the user-specified definition of γ and φ, a ramp is sometimes detected by the L1-SW but not by the multi-parameter segmentation algorithm. For the same reason, the multi-parameter segmentation algorithm may detect multiple smaller ramps, while the L1-SW combines these smaller ramps into one larger ramp.

For the case of $\lambda = 0.5, \gamma = 1 \times 10^{-4}$, the multi-parameter segmentation algorithm detected multiple smaller ramps, while the L1-SW detected fewer ramps with a large magnitude and duration which extends these smaller ramps. The ramps of the L1-SW do not closely approximate the signal. One ramp was detected by the multi-parameter segmentation algorithm but missed by the L1-SW.

Furthermore, in both Fig. 78 and Fig. 79, the multi-parameter segmentation algorithm performs better than the L1-SW with regards to the determination of start- and end-points of ramps, i.e. the detected ramps of the multi-parameter segmentation algorithm always end on a local minima or maxima, whereas the detected ramps of the L1-SW do not. Additionally, the ramps detected by the L1-SW are not anchored to the points of the signal, contrary to the ramps of the multi-parameter segmentation algorithm. The ramps detected by the L1-SW are also sometimes interjected by a horizontal segment which wrongly extends the length of the significant ramp. Conversely, the multi-parameter segmentation algorithm discards these horizontal segments from the significant ramps. The sets of ramps of both methods correspond well to the visually detected ramps, however the ramps depend on the preference of the user.

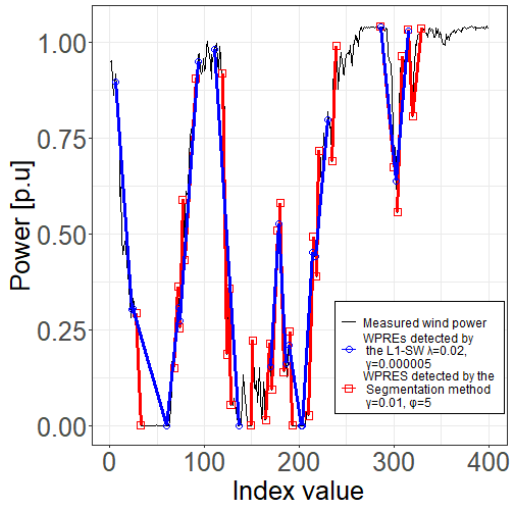


Fig. 78: A comparison of the significant ramps detected by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ to the significant ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.1, \varphi = 5$.

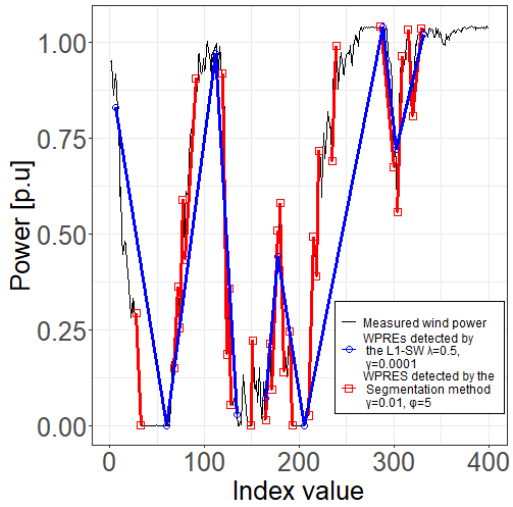


Fig. 79: A comparison of the significant ramps detected by the L1-SW for $\lambda = 0.5, \gamma = 1 \times 10^{-4}$ to the significant ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.1, \varphi = 5$.

3.9.3 Run-time and number of upward and downward ramps

Fig. 80 shows the number of ramps detected by each method for given parameter values, and the proportion of up and down ramps for each method is also indicated. It is evident that the multi-parameter segmentation algorithm detects the most ramps among all the methods, and the number of detected ramps is also more robust to parameter changes, when compared to the other methods. The SDA detected the least ramps among all the methods, and majority of the ramps are upward ramps. Conversely, the number of upward ramps versus downward ramps are approximately equal for the multi-parameter segmentation algorithm, OpSDA and the L1-SW.

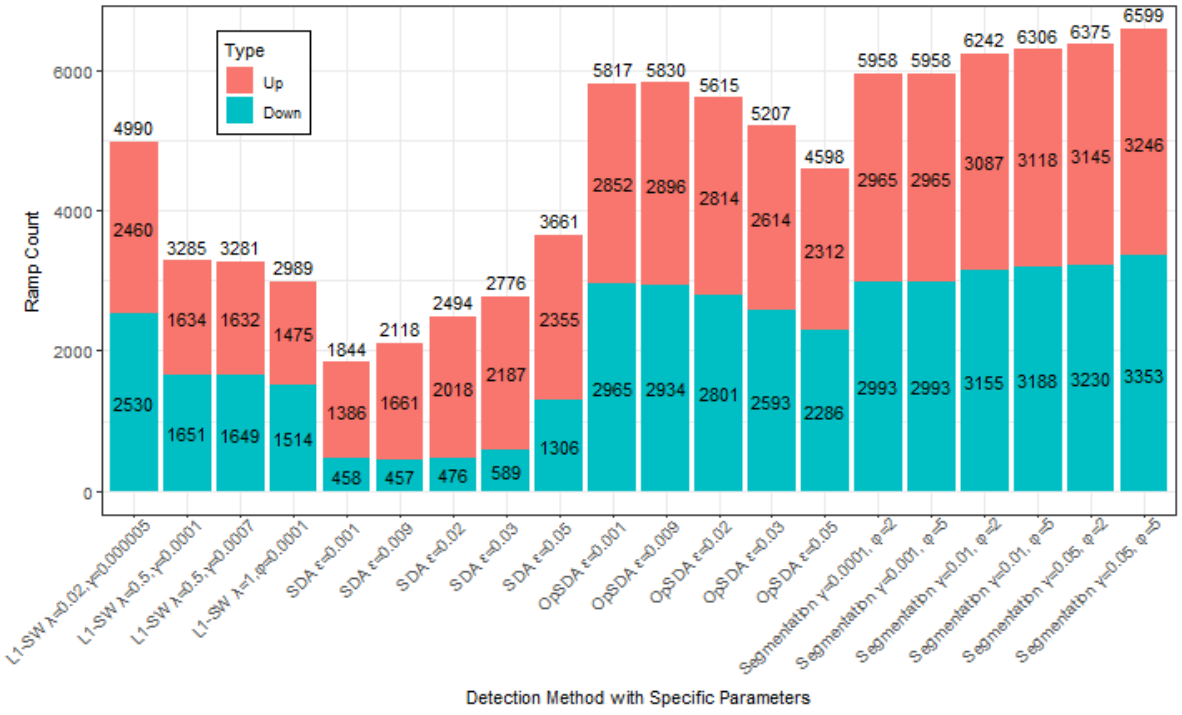


Fig. 80: Number of upward and downward ramps detected by the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for various parameter values.

Fig. 81 shows the run-time of the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for various parameter values. It is recognised that the performance of the SDA and the multi-parameter segmentation algorithm is superior to that of the L1-SW and OpSDA based on run-time. The SDA and the multi-parameter segmentation algorithm perform similarly based on run-time, however, the multi-parameter segmentation algorithm outperforms the SDA for smaller values of epsilon.

The dynamic recursion technique causes the lengthy run-time of the L1-SW and OpSDA. The difference in run-times between the OpSDA and L1-SW can be attributed to the number of data points inputted into the dynamic recursion technique, which is shown in Table 11 as reference. The run-time is directly proportional to the number of data points inputted into the dynamic programming recursion. The L1-SW uses the L1-trend filtering technique to reduce the size of the dataset, while the OpSDA uses the SDA. Furthermore, since the OpSDA utilises the SDA to segregate the wind power signal before the dynamic programming recursion is implemented, the run-time of the SDA is also included in the run-time of the OpSDA.

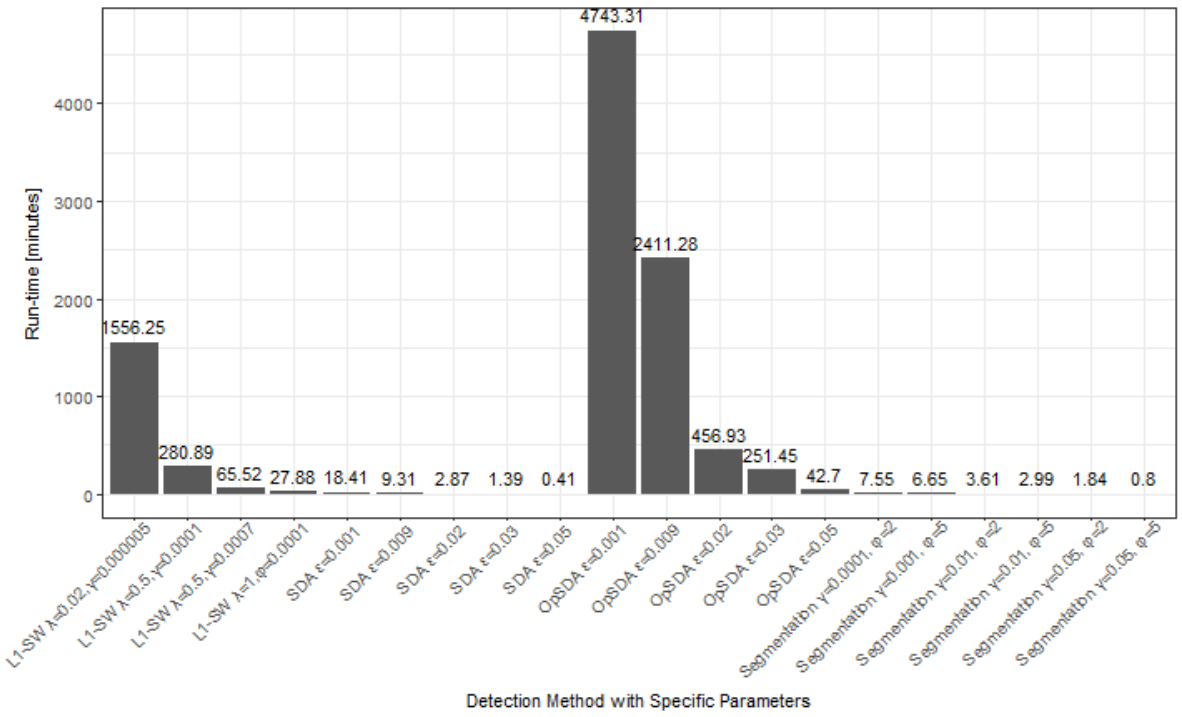


Fig. 81: Run-time of the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for various parameter values.

Table 11: The number of points in the reduced dataset of the OpSDA and L1-SW for various parameters, which are inputted into the dynamic recursion technique to merge adjacent segments with the same direction.

SDA		**L1-SW**	
Parameters	**Number of points in the reduced dataset**	**Parameters**	**Number of points in the reduced dataset**
$\varepsilon = 0.001$	88961	$\lambda = 0.02, \gamma = 5 \times 10^{-6}$	61167
$\varepsilon = 0.009$	62611	$\lambda = 0.5, \gamma = 1 \times 10^{-4}$	20876
$\varepsilon = 0.02$	42217	$\lambda = 0.5, \gamma = 7 \times 10^{-4}$	19179
$\varepsilon = 0.03$	31066	$\lambda = 1, \gamma = 1 \times 10^{-4}$	16921
$\varepsilon = 0.05$	19029		

3.9.4 Ramp comparison based on a mathematical test

The set of ramps identified by the multi-parameter segmentation algorithm for $\gamma = 0.01$, $\varphi = 5$ are compared to the sets of ramps detected by the SDA and OpSDA for $\varepsilon = 9 \times 10^{-3}$, as well as the sets of ramps detected by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ and $\lambda = 0.5, \gamma = 1 \times 10^{-4}$ respectively, to identify the ramps that are equivalent. Two ramps are considered equivalent if the length of the overlapping intervals between the two is greater than or equal to a chosen threshold. The threshold is chosen as 80% of the mean length of the two intervals. It is recognised that adjusting the threshold length of the overlapping interval will result in different equivalent ramps being identified. Furthermore, only one ramp from the first method can be equivalent to one ramp of the second method. Based on this criteria, a contingency table comparing the ramps detected by two ramp detection methods is constructed

by identifying the ramps detected by both methods, as well as the ramps detected by each method individually. Table 12 to Table 15 depict the contingency tables that resulted when the significant ramps extracted by the multi-parameter segmentation algorithm are compared to the significant ramps extracted via the SDA, OpSDA and the L1-SW respectively, for optimal tunable parameter values and the selected ramp definition.

Table 12 shows the contingency table which resulted from comparing the set of ramps detected by the SDA for $\varepsilon = 9 \times 10^{-3}$ to the set of ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \varphi = 5$. The total significant ramps detected by the multi-parameter segmentation algorithm or the SDA is 7891. The total significant ramps consist of 6306 significant ramps detected by the multi-parameter segmentation algorithm and 2118 significant ramps detected by the SDA, of which a total of 533 ramps are detected by both the multi-parameter segmentation algorithm and the SDA. The number of significant ramps detected by the SDA but not detected by the multi-parameter segmentation algorithm is 1585, and the number of significant ramps detected by the multi-parameter segmentation algorithm but not detected by the SDA is 5773.

Table 12: Comparison of the ramps detected by the SDA for $\varepsilon = 9 \times 10^{-3}$ to the ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \varphi = 5$.

	Multi-parameter segmentation algorithm (Yes)	**Multi-parameter segmentation algorithm (No)**	**Total**
SDA (Yes)	533	1585	2118
SDA (No)	5773	0	5773
Total	6306	1585	7891

Table 13 shows the contingency table which resulted from comparing the set of ramps detected by the OpSDA with $\varepsilon = 9 \times 10^{-3}$ to the set of ramps detected by the multi-parameter segmentation algorithm with $\gamma = 0.01,\ \varphi = 5$. The total significant ramps detected by the multi-parameter segmentation algorithm or the OpSDA is 8434. The total significant ramps consist of 6306 significant ramps detected by the multi-parameter segmentation algorithm and 5830 significant ramps detected by the OpSDA, of which a total of 3702 ramps are detected by both the multi-parameter segmentation algorithm and the OpSDA. The number of significant ramps detected by the OpSDA but not detected by the multi-parameter segmentation algorithm is 2128, and the number of significant ramps detected by the multi-parameter segmentation algorithm but not detected by the OpSDA is 2604.

Table 13: Comparison of the ramps detected by the OpSDA and $\varepsilon = 9 \times 10^{-3}$ to the ramps detected by the multi-parameter segmentation algorithm with $\gamma = 0.01,\ \varphi = 5$.

	Multi-parameter segmentation algorithm (Yes)	**Multi-parameter segmentation algorithm (No)**	**Total**
OpSDA (Yes)	3702	2128	5830
OpSDA (No)	2604	0	2604
Total	6306	2128	8434

Table 14 shows the contingency table which resulted from comparing the set of ramps detected by the L1-SW with $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ to the set of ramps detected by the multi-parameter segmentation algorithm with $\gamma = 0.01,\ \varphi = 5$. The total significant ramps detected by the multi-parameter segmentation algorithm or the L1-SW is 9231. The total significant ramps consist of 6306 significant ramps detected by the multi-parameter segmentation algorithm and 4990 significant ramps detected by the L1-

SW, of which a total of 2065 ramps are detected by both the multi-parameter segmentation algorithm and the L1-SW. The number of significant ramps detected by the L1-SW but not detected by the multi-parameter segmentation algorithm is 2925, and the number of significant ramps detected by the multi-parameter segmentation algorithm but not detected by the L1-SW is 4241.

Table 14: Comparison of the ramps detected by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ to the ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \varphi = 5$.

	Multi-parameter segmentation algorithm (Yes)	Multi-parameter segmentation algorithm (No)	Total
L1-SW (Yes)	2065	2925	4990
L1-SW (No)	4241	0	4241
Total	6306	2925	9231

Table 15 shows the contingency table which resulted from comparing the set of ramps detected by the L1-SW with $\lambda = 0.5, \gamma = 1 \times 10^{-4}$ to the set of ramps detected by the multi-parameter segmentation algorithm with $\gamma = 0.01,\ \varphi = 5$. The total significant ramps detected by the multi-parameter segmentation algorithm or the L1-SW is 9161. The total significant ramps consist of 6306 significant ramps detected by the multi-parameter segmentation algorithm and 3285 significant ramps detected by the L1-SW, of which a total of 430 ramps are detected by both the multi-parameter segmentation algorithm and the L1-SW. The number of significant ramps detected by the L1-SW but not detected by the multi-parameter segmentation algorithm is 2855, and the number of significant ramps detected by the multi-parameter segmentation algorithm but not detected by the L1-SW is 5876.

Table 15: Comparison of the ramps detected by the L1-SW for $\lambda = 0.5, \gamma = 1 \times 10^{-4}$ to the ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \varphi = 5$.

	Multi-parameter segmentation algorithm (Yes)	Multi-parameter segmentation algorithm (No)	Total
L1-SW (Yes)	*430*	*2855*	*3285*
L1-SW (No)	*5876*	*0*	*5876*
Total	*6306*	*2855*	*9161*

Furthermore, the suite of metrics listed in (24) to (27) is calculated for each contingency table in Table 12 to Table 15 to evaluate how the ramp detection performance of the multi-parameter segmentation algorithm compares to the ramp detection performance of the SDA, the OpSDA and the L1-SW for optimal parameter values respectively, and the corresponding results are summarised in Table 16. The suite of metrics includes the Probability Of Detection (POD), Critical Success Index (CSI), Frequency Bias Score (FBS) and the Success Ratio (SR). The POD represents the fraction of the ramp events detected by the multi-parameter segmentation algorithm that are also detected by the second method. SR represents the fraction of the ramp events detected by the second method that are not detected by the multi-parameter segmentation algorithm. CSI represents the fraction of ramp events detected by the second method or the multi-parameter segmentation algorithm that are detected by both methods. The CSI ranges from 0 to 1, where 0 represents the worst possible correspondence and 1 represents the best correspondence. The FBS represents the ratio of the ramp events detected by the second method to the ramp events detected by the multi-parameter segmentation algorithm. If the FBS has a value smaller than one, the number of ramp events detected by the multi-parameter segmentation algorithm exceeds the number of ramp events detected by the second method. Conversely, if the FBS is greater than one, the number of

ramp events detected by the second method exceeds the number of ramp events detected by the multi-parameter segmentation algorithm.

From Table 16, it is evident that the fraction of the ramp events detected by the multi-parameter segmentation algorithm that are also detected by the SDA is 8.4523%, i.e. the fraction of the ramp events detected by the multi-parameter segmentation algorithm that are not detected by the SDA is 91.5477%. The fraction of the ramp events detected by the SDA that are not detected by the multi-parameter segmentation algorithm is 74.8347%, i.e. the fraction of the ramp events detected by the SDA that are also detected by the multi-parameter segmentation algorithm is 25.1653%. The fraction of the ramp events detected by the SDA or the multi-parameter segmentation algorithm that are detected by both methods is 6.7545%. The ratio of the ramp events detected by the SDA to the ramp events detected by the multi-parameter segmentation algorithm is 0.3359, therefore, the number of ramp events detected by the multi-parameter segmentation algorithm exceeds the number of ramp events detected by the SDA.

From Table 16, it is evident that the fraction of the ramp events detected by the multi-parameter segmentation algorithm that are also detected by the OpSDA is 58.706%, i.e. the fraction of the ramp events detected by the multi-parameter segmentation algorithm that are not detected by the OpSDA is 41.294%. The fraction of the ramp events detected by the OpSDA that are not detected by the multi-parameter segmentation algorithm is 36.5009%, i.e. the fraction of the ramp events detected by the OpSDA that are also detected by the multi-parameter segmentation algorithm 63.4991%. The fraction of the ramp events detected by the OpSDA or the multi-parameter segmentation algorithm that are detected by both methods is 43.8938%. The ratio of the ramp events detected by the OpSDA to the ramp events detected by the multi-parameter segmentation algorithm is 0.9245, therefore, the number of ramp events detected by the multi-parameter segmentation algorithm slightly exceeds the number of ramp events detected by the OpSDA.

It is evident that the fraction of the ramp events detected by the multi-parameter segmentation algorithm that are also detected by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ is 32.7466%, i.e. 67.2534% of the ramp events detected by the multi-parameter segmentation algorithm are not detected by the L1-SW. The fraction of the ramp events detected by the L1-SW that are not detected by the multi-parameter segmentation algorithm is 58.6172%, i.e. 41.3828% of the ramp events detected by the L1-SW are also detected by the multi-parameter segmentation algorithm. The fraction of the ramp events detected by the L1-SW or the multi-parameter segmentation algorithm that are detected by both methods is 22.3703%. The ratio of the ramp events detected by the L1-SW to the ramp events detected by the multi-parameter segmentation algorithm is 0.7913, therefore, the number of ramp events detected by the multi-parameter segmentation algorithm exceeds the number of ramp events detected by the L1-SW.

It is evident that the fraction of the ramp events detected by the multi-parameter segmentation algorithm that are also detected by the L1-SW with $\lambda = 0.5, \gamma = 1 \times 10^{-4}$ is 6.8189% , i.e. 93.1811% of the ramp events detected by the multi-parameter segmentation algorithm are not detected by the L1-SW. The fraction of the ramp events detected by the L1-SW that are not detected by the multi-parameter segmentation algorithm is 86.9102%, i.e. 13.0898% of the ramp events detected by the L1-SW are also detected by the multi-parameter segmentation algorithm. The fraction of the ramp events detected by the L1-SW or the multi-parameter segmentation algorithm that are detected by both methods is 4.6938%. The ratio of the ramp events detected by the L1-SW to the ramp events detected by the multi-parameter

segmentation algorithm is 0.5209, therefore, the number of ramp events detected by the multi-parameter segmentation algorithm exceeds the number of ramp events detected by the L1-SW.

Furthermore, the CSI metric is the highest when comparing the ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \varphi = 5$ to the ramps detected by the OpSDA for $\varepsilon = 9 \times 10^{-3}$. Therefore, the set of ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \varphi = 5$ is the most similar to the set of ramps detected by the OpSDA for $\varepsilon = 9 \times 10^{-3}$. The CSI metric is the lowest when comparing the ramps detected by the multi-parameter segmentation algorithm with $\gamma = 0.01,\ \varphi = 5$ to the ramps detected by the L1-SW with $\lambda = 0.5, \gamma = 1 \times 10^{-4}$. The set of ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \varphi = 5$ is, therefore, the least similar to the set of ramps detected by the L1-SW for $\lambda = 0.5, \gamma = 1 \times 10^{-4}$.

Table 16: Comparison of the set of ramps detected by the multi-parameter segmentation algorithm to the sets of ramps detected by the SDA, OpSDA and the L1-SW for optimal tunable parameter values.

Method compared to the multi-parameter segmentation algorithm with $\boldsymbol{\gamma = 0.01,\ \varphi = 5}$	POD	SR	CSI	FBIAS
SDA with $\boldsymbol{\varepsilon = 9 \times 10^{-3}}$	8.4523%	74.8347%	6.7545%	0.3359
OpSDA with $\boldsymbol{\varepsilon = 9 \times 10^{-3}}$	58.706%	36.5009%	43.8938%	0.9245
L1-SW with $\boldsymbol{\lambda = 0.02, \gamma = 5 \times 10^{-6}}$	32.7466%	58.6172%	22.3703%	0.7913
L1=SW with $\boldsymbol{\lambda = 0.5, \gamma = 1 \times 10^{-4}}$	6.8189%	86.9102%	4.6938%	0.5209

3.9.5 Detection accuracy of the ramp start- and end-points

The metrics listed in (13) and (14) are used to evaluate and compare the ramp detection performance of the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values. Specifically, these metrics are used to determine whether the start- or end-point of a given Wind Power Ramp Event (WPRE) is extracted successfully or not. For the WPREs detected using the L1-SW, these metrics were evaluated based on the actual signal as well as the L1-trend filtered signal. This information was included to account for the fact that the ramps detected by the L1-SW were identified based on the L1-trend filtered signal and not the actual signal.

Table 23 to Table 27 in Appendix A show the contingency tables listing the number of wind power ramp events with start-Yes-end-Yes (sYeY), start-Yes-end-No (sYeN), start-No-end-Yes (sNeY) and start-No-end-No (sNeN) identified by the SDA, OpSDA, and the L1-SW, based on evaluating the metrics according to both the actual signal and L1-trend filtered signal, as well as the multi-parameter segmentation algorithm, respectively, for optimal parameter values. The number of ramps with an accurate start- and end-point is denoted by sYeY. The number of detected ramps with accurate start-points, but inaccurate end-points are represented by sYeN. The number of detected ramps with inaccurate start-points, but accurate end-points are represented by sNeY. The number of detected ramps with inaccurate start- and end-points are denoted by sNeN [20].The corresponding percentage out of the total number of wind power ramp events is also indicated.

The results for Table 23 to Table 27 in Appendix A is visualised via pie charts in Fig. 82 to Fig. 88, to facilitate the comparison of the ramp detection performance of the various methods based on the detection accuracy of the ramp start- and end-points. Specifically, Fig. 82 and Fig. 83 illustrate the percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the SDA and OpSDA, respectively, for $\varepsilon = 0.009$. Fig. 84 and Fig. 85 illustrate the percentage of ramps with sYeY, sYeN,

sNeY and sNeN out of the total wind power ramp events identified by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ and $\lambda = 0.5, \gamma = 1 \times 10^{-4}$, respectively, when the metrics are evaluated based on the actual signal. Fig. 86 and Fig. 87 illustrate the percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the L1-SW for $\lambda = 0.02, \gamma = 5 \times 10^{-6}$ and $\lambda = 0.5, \gamma = 1 \times 10^{-4}$, respectively, when the metrics are evaluated based on the L1-trend filtered signal. Fig. 88 illustrates the percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the multi-parameter segmentation algorithm for $\gamma = 0.1, \varphi = 5$.

The percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events is indicated in purple, blue, green and pink, respectively. It is evident that the multi-parameter segmentation algorithm successfully extracts the start-point and end-point of all the detected WPREs and is superior to the other methods with regards to the detection accuracy of the start- and end-points of the ramps. When testing the metrics in (13) and (14) for the L1-SW based on the L1-trend filtered signal, the percentage of ramps with accurate start- and end-point is also very high, above 96%. However, when testing the metrics in (13) and (14) for the L1-SW based on the actual signal, the percentage of ramps with accurate start- and end-point is very low, less than 30%.

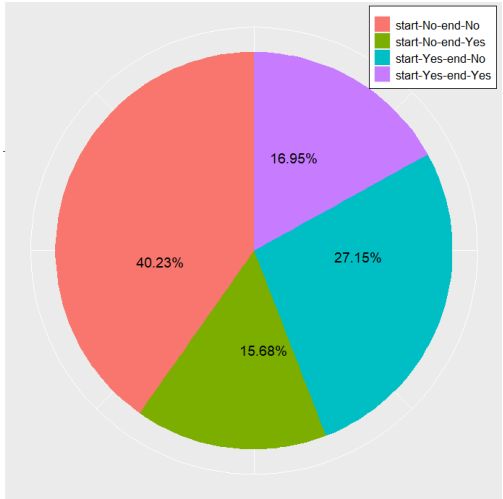


Fig. 82: The percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the SDA for $\varepsilon = 0.009$.

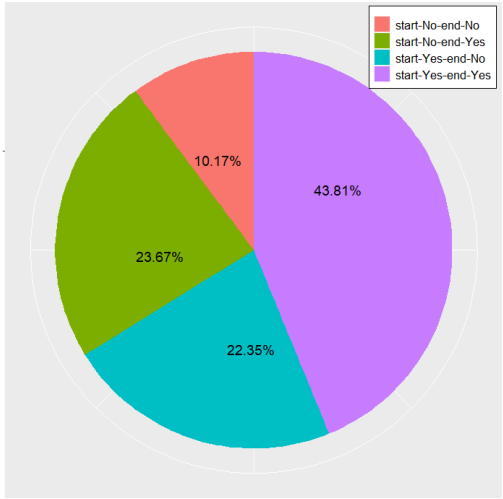


Fig. 83: The percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the OpSDA for $\varepsilon = 0.009$.

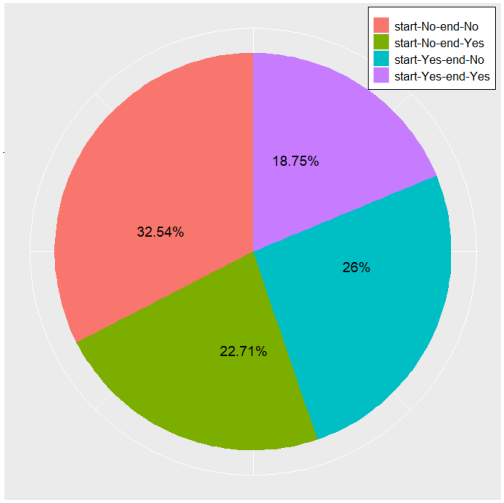


Fig. 84: The percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the L1-SW for $\lambda = 0.5,\ \gamma = 1 \times 10^{-4}$.

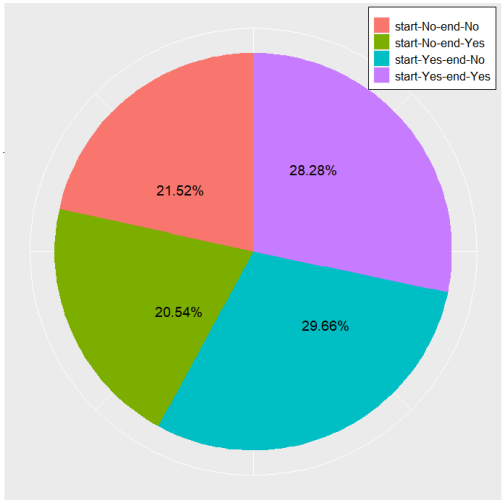


Fig. 85: The percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the L1-SW for $\lambda = 0.02,\ \gamma = 5 \times 10^{-6}$.

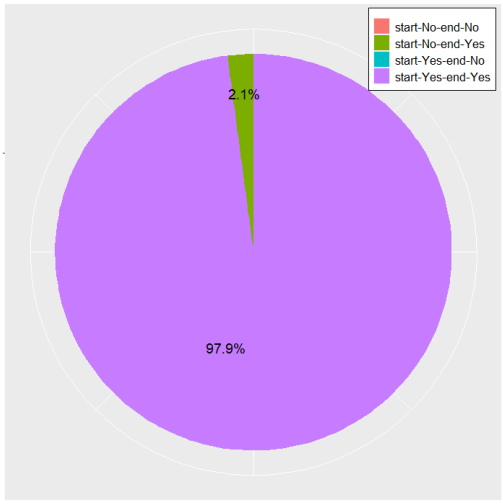


Fig. 86: The percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the L1-SW for $\lambda = 0.5,\ \gamma = 1 \times 10^{-4}$, based on the L1-trend filtered signal.

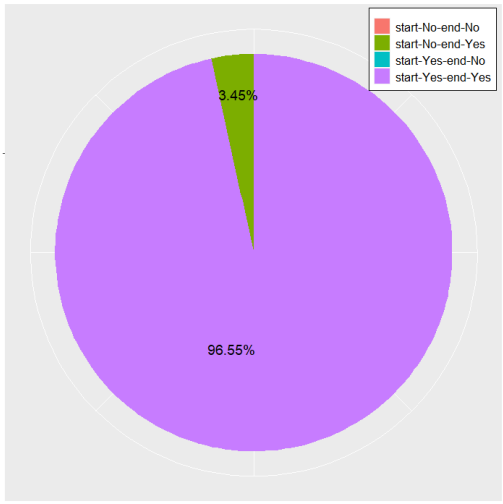


Fig. 87: The percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the L1-SW for $\lambda = 0.02,\ \gamma = 5 \times 10^{-6}$, based on the L1-trend filtered signal.

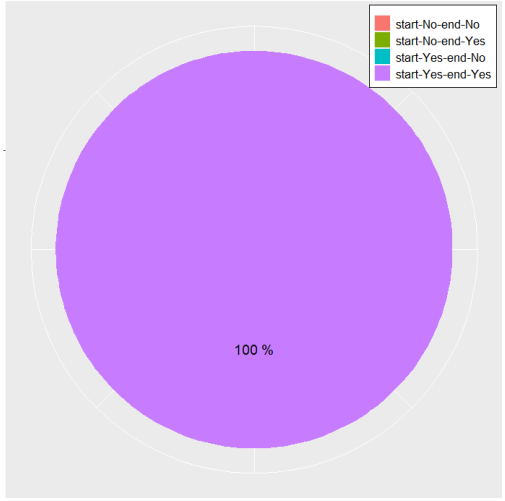


Fig. 88: The percentage of ramps with sYeY, sYeN, sNeY and sNeN out of the total wind power ramp events identified by the multi-parameter segmentation algorithm for $\gamma = 0.01, \varphi = 5$.

Furthermore, a suite of metrics listed in (17)-(20), namely Probability of Detection (POD), Critical Success Index (CSI), Frequency Bias Score (FBS) and Success Ratio (SR), are calculated for each contingency table in Table 23 to Table 27 to evaluate and compare the ramp detection performance of the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values, respectively. POD represents the fraction of the detected ramp events with accurate end-points that have accurate start-points. SR represents the fraction of the detected ramp events with accurate start-points that have accurate end-points. CSI represents the fraction of ramp events with accurate ramp start or accurate ramp end that have both an accurate ramp start and ramp end. The CSI ranges from 0 to 1, where 0 represents the worst detection accuracy and 1 represents the best detection accuracy. The FBS represents the ratio of the ramp events with accurate start-points to the ramp events with accurate end-points. If the FBS has a value smaller than one, the number of ramp events with accurate end-points exceeds the number of ramp events with accurate start-points. Conversely, if the FBS is greater than one, the number of ramp events with accurate start-points exceeds the number of ramp events with accurate end-points.

The obtained results are listed in Table 17 and visualised in the performance diagram in Fig. 89. The x-axis of the performance diagram represents the SR, and the y-axis represents the POD. Additionally, the diagonal dashed lines represents the FBS, and the curved lines represent the CSI [54]. The closer a point representing the ramp detection results of a respective method for given parameters is to the top right corner of the performance diagram, the better the ramp detection performance. Only the results of the suite of metrics based on the actual signal is included in the performance diagram. The performance diagram illustrates the ramp detection results for the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for various parameter values, besides the parameters used in Table 17.

From Table 17 and Fig. 89, it is evident that the ramp detection performance of the multi-parameter segmentation algorithm for all the various parameter values is superior to the ramp detection performance of the SDA, OpSDA and the L1-SW. Specifically, the multi-parameter segmentation algorithm

has a CSI of 1 which indicates the best detection accuracy, and the points representing the multi-parameter segmentation algorithm in the performance diagram is situated in the top right corner. The OpSDA has the second-best ramp detection performance based the suite of metrics, and the SDA and L1-SW perform similarly.

Table 17: Suite of metrics illustrating the ramp detection performance of the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm based on the detection accuracy of the start- and end-points.

Method	Parameters	POD	CSI	FBS	SR
SDA	$\varepsilon = 9 \times 10^{-3}$	0.5195	0.2836	1.3517	0.3844
OpSDA	$\varepsilon = 9 \times 10^{-3}$	0.6492	0.48778	0.9804	0.6622
L1-SW	$\lambda = 0.02, \quad \gamma = 5 \times 10^{-6}$	0.5792	0.3603	1.1868	0.4881
For actual signal	$\lambda = 0.5, \quad \gamma = 1 \times 10^{-4}$	0.4523	0.2780	1.0793	0.4190
L1-SW	$\lambda = 0.02, \quad \gamma = 5 \times 10^{-6}$	0.9790	0.9790	0.9790	1
For L1-trend filtered signal	$\lambda = 0.5, \quad \gamma = 1 \times 10^{-4}$	0.9679	0.9543	0.9821	0.9855
Multi-parameter segmentation algorithm	$\gamma = 0.01, \quad \lambda = 5$	1	1	1	1

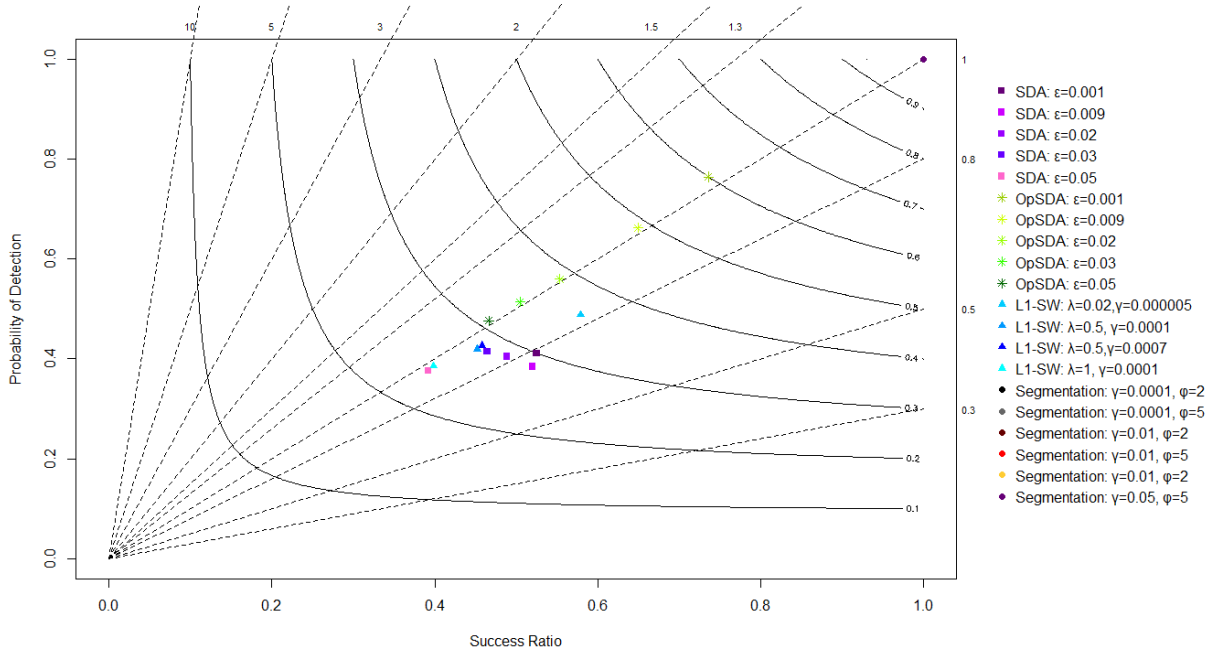


Fig. 89: Performance diagram evaluating the ramp detection performance of the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for various parameter values based on the detection accuracy of the start- and end-points of the ramps.

3.9.6 Comparison of key ramp features

3.9.6.1 Overview

The wind power ramps identified by the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values are used to perform statistical analysis of the key ramp features. Specifically, the probability density functions are computed for the ramp duration and the ramp magnitude of the upward and downward ramps detected using the SDA, OpSDA, L1-SW and the multi-

parameter segmentation algorithm, respectively. The aim of this analysis is to statistically compare the key features of the ramps extracted by each method.

3.9.6.2 Ramp duration

Fig. 90 and Fig. 91 depict the Probability Density Functions (PDFs) of the ramp duration of the upward and downward ramps, respectively, detected using (i) the SDA with $\varepsilon = 0.009$, (ii) the OpSDA with $\varepsilon = 0.009$, (iii) the L1-SW with $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$, (iv) the L1-SW with $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$ and (v) multi-parameter segmentation algorithm with $\gamma = 0.01, \varphi = 5$. Furthermore, Table 18 summarises the mean-, maximum-, minimum- and 3rd quartile-value observed for the ramp duration of the upward and downward ramps detected by (i)-(v), respectively, to support Fig. 90 and Fig. 91. A detailed discussion of these results was also presented in section 3.3.3, section 3.4.3, section 3.5.3 and section 3.6.3, respectively.

Firstly, it is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by each method will have a short duration rather than a long duration. From Fig. 90 and Fig. 91, as well as Table 18, it is evident that the ramps detected using the SDA for $\varepsilon = 0.009$ has a shorter duration than the ramps detected by the OpSDA, L1-SW and the multi-parameter segmentation algorithm, considering that the SDA does not merge adjacent ramps with the same direction while the other methods do. The ramps detected by the L1-SW with $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$ has a longer duration than the ramps detected by the SDA, OpSDA and multi-parameter segmentation algorithm as well as the L1-SW with $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$, which is a consequence of the size of λ. Furthermore, the probability density functions for the L1-SW with $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$, the OpSDA with $\varepsilon = 0.009$ and the multi-parameter segmentation algorithm display a high degree of similarity, signifying that the ramps extracted by these methods for the given parameter values have similar durations. Nevertheless, the mean duration observed for the ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01, \varphi = 5$ is shorter than the mean duration observed for the ramps detected by the OpSDA and L1-SW, and the maximum duration observed for the ramps detected by the OpSDA and L1-SW for $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$ far exceeds the maximum duration observed for the ramps detected by the multi-parameter segmentation algorithm. A possible reason for this includes that the ramps detected by the OpSDA and L1-SW are sometimes interjected by a horizontal segment which wrongly extends the length of the significant ramp, while the multi-parameter segmentation algorithm employs post processing to discard the horizontal segments. A high degree of symmetry is observed in the PDFs of the ramp duration between the upward and downward ramps, which implies that the upward and downward ramps have similar durations.

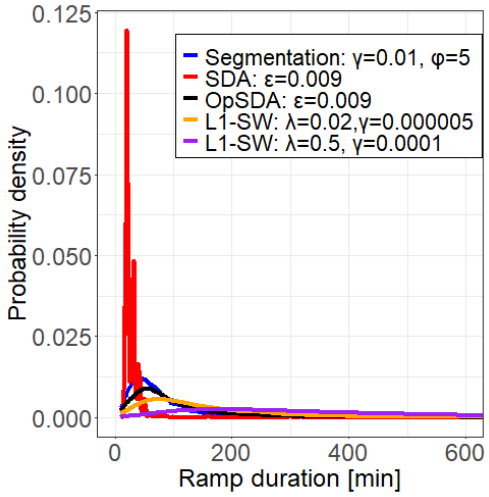


Fig. 90: Probability density function of the ramp duration of upward ramps detected by the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values.

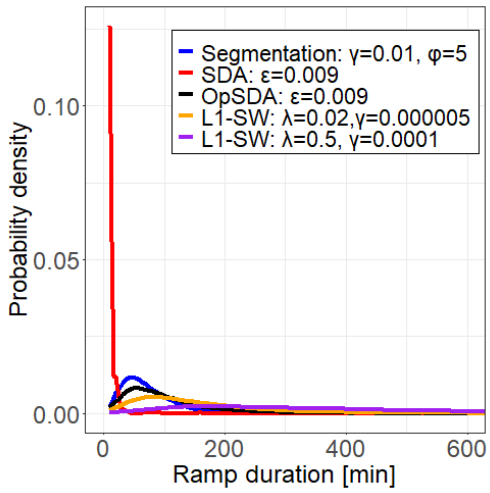


Fig. 91: Probability density function of the ramp duration of downward ramps detected by the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values.

Table 18: Summary statistics of the PDFs of the ramp duration of the upward and downward ramps, respectively, detected by the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values.

Method	Parameters	Mean ramp duration [min]		Maximum ramp duration [min]		Minimum ramp duration [min]		3rd Quartile value of ramp duration [min]	
		Up	Down	Up	Down	Up	Down	Up	Down
SDA	$\varepsilon = 9 \times 10^{-3}$	28.72	12.34	740	160	20	10	30	10
OpSDA	$\varepsilon = 9 \times 10^{-3}$	112.32	104.83	1180	1050	20	10	130	130
L1-SW	$\lambda = 0.02,$ $\gamma = 5 \times 10^{-6}$	147.76	158.11	1130	1230	10	10	190	200
	$\lambda = 0.5,$ $\gamma = 1 \times 10^{-4}$	327.59	372.64	1370	1830	40	50	440	490
Multi-parameter segmentation algorithm	$\gamma = 0.01,$ $\lambda = 5$	70.98	73.31	400	400	10	10	90	90

3.9.6.3 Ramp magnitude

Fig. 92 and Fig. 93 depict the Probability Density Functions (PDFs) of the ramp magnitude of the upward and downward ramps, respectively, detected using (i) the SDA with $\varepsilon = 0.009$, (ii) the OpSDA with $\varepsilon = 0.009$, (iii) the L1-SW with $\lambda = 0.02,\ \gamma = 5 \times 10^{-6}$, (iv) the L1-SW with $\lambda = 0.5,\ \gamma = 1 \times 10^{-4}$and (v) the multi-parameter segmentation algorithm with $\gamma = 0.01, \varphi = 5$. Furthermore, Table 19 summarises the mean-, maximum-, minimum- and 3rd quartile-value observed for the ramp magnitude of the upward and downward ramps detected by (i)-(v), to support Fig. 92 and Fig. 93. A detailed discussion of these results was included in section 3.3.3, section 3.4.3, section 3.5.3 and section 3.6.3, respectively.

Firstly, it is evident that all the PDFs for the upward and downward ramps are right skewed, indicating that it is more likely that the ramps detected by the various methods will have a small magnitude rather

than a large magnitude. From Fig. 92 and Fig. 93, as well as Table 19, it is evident that the ramps detected using the SDA with $\varepsilon = 0.009$ has a smaller magnitude than the ramps detected by the OpSDA, L1-SW and the multi-parameter segmentation algorithm, considering that the SDA does not merge adjacent ramps with the same direction while the other methods do. The ramps detected by the L1-SW with $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$ has a larger magnitude than the ramps detected by the SDA, OpSDA and the multi-parameter segmentation algorithm as well as the L1-SW with $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$, which is a consequence of the size of λ. Furthermore, the probability density functions for the L1-SW with $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$, the OpSDA with $\varepsilon = 0.009$ and the multi-parameter segmentation algorithm for $\gamma = 0.01, \varphi = 5$ display a high degree of similarity, signifying that the ramps extracted by these methods for the given parameter values have similar magnitude. A high degree of symmetry is observed in the PDFs of the ramp magnitude between the upward and downward ramps, which implies that the upward and downward ramps have similar magnitudes. Additionally, it is recognised that the minimum magnitude observed for the ramps detected by (i)-(v) is 0.20 p.u. This can be explained by the minimum value chosen for the ramp definition, i.e. a significant ramp is defined as a change in wind power output that is greater than or equal to 20% of the maximum wind power output without restricting the ramp duration.

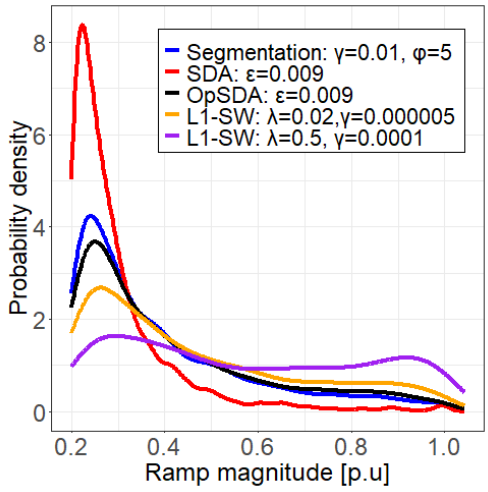


Fig. 92: Probability density function of the ramp magnitude of upward ramps detected by the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values.

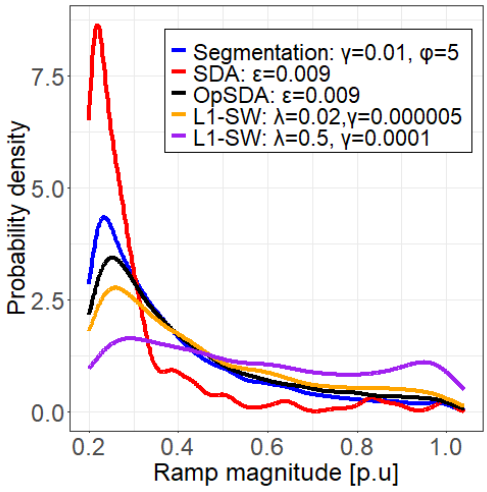


Fig. 93: Probability density function of the ramp magnitude of downward ramps detected by the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values.

Table 19: Summary statistics of the PDFs of the ramp magnitude of the upward and downward ramps detected by the SDA, OpSDA, L1-SW and the multi-parameter segmentation algorithm for optimal parameter values.

Method	Parameters	Mean ramp magnitude [p.u.]		Maximum ramp magnitude [p.u.]		Minimum ramp magnitude [p.u.]		3rd Quartile value of ramp magnitude [p.u.]	
		Up	Down	Up	Down	Up	Down	Up	Down
SDA	$\varepsilon = 9 \times 10^{-3}$	0.2984	0.3028	1.0041	1.0041	0.2	0.2	0.3214	0.3083
OpSDA	$\varepsilon = 9 \times 10^{-3}$	0.4112	0.4146	1.0403	1.0401	0.2	0.2	0.5066	0.5107
L1-SW	$\lambda = 0.02,$ $\gamma = 5 \times 10^{-6}$	0.4626	0.4507	1.0433	1.0406	0.2	0.2	0.6057	0.5815
	$\lambda = 0.5,$ $\gamma = 1 \times 10^{-4}$	0.5715	0.566	1.0409	1.0395	0.2	0.2	0.8047	0.7881
Multi-parameter segmentation algorithm	$\gamma = 0.01,$ $\lambda = 5$	0.3922	0.3835	1.0432	1.0395	0.2	0.2	0.4772	0.4614

4 Statistical analysis

4.1 Overview

In this chapter various statistics are presented to analyse the ramps detected via the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$, in order to gain valuable insights into the wind power ramp events and the characteristics of its features which may prove to be helpful in making informed scheduling decisions.

4.2 Statistics of the detected wind power ramps

Fig. 94 and Fig. 95 show scatterplots of the joint distribution of the duration and magnitude of upward ramps and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$, respectively. Each point on the scatterplot represents a single Wind Power Ramp Event (WPRE), and the horizontal and vertical position of the point indicate the duration and magnitude of the WPRE, respectively. The relationship between the magnitude and duration of the detected ramps can be observed from Fig. 94 and Fig. 95. A trend line was also added to both scatterplots to facilitate the deduction of the correlational relationship between the magnitude and duration of upward and downward ramps, respectively. Furthermore, the points were made partially transparent to address overplotting, and the marginal distribution of the ramp magnitude and ramp duration is also included.

From Fig. 94 and Fig. 95, and specifically the trend lines, it is evident that a weak positive linear relationship exists between the ramp duration and ramp magnitude. Correspondingly, the ramp magnitude is directly proportional to the ramp duration. It is important to note that causality is not implied for the variables of interest. In both Fig. 94 and Fig. 95, most of the points are situated in the bottom left corner of the plot. Most of the detected ramps, therefore, have a short duration and small magnitude, which do not present great concern to the power system operators. Specifically, based on visual inspection, most of the detected ramps have a duration shorter than approximately 120 minutes and a magnitude between approximately 20 MW and 50 MW. The marginal distributions of the ramp magnitude and ramp duration also reflects this information. The minimum value observed for the ramp magnitude is 20 MW, which corresponds to the chosen ramp definition, i.e. a significant ramp is defined as a change in wind power output that is greater than or equal to 20% of the wind power capacity (100 MW) without restricting the ramp duration. The points in the top left corner of the plot, i.e. ramps with large magnitude and short duration which presents concern to system operators, are sparsely dispersed and does not occur frequently. A few points are situated on the right-side of the plot, therefore, the multi-parameter segmentation algorithm did not detect a lot of ramps with a duration larger than approximately 200 minutes. Furthermore, a high degree of symmetry is observed in the scatterplots of the joint distribution of the ramp magnitude and ramp duration for the upward and downward ramps, as well as the respective marginal distributions. The upward and downward ramps, therefore, have similar duration and magnitude.

Furthermore, Fig. 96 and Fig. 97 show the joint distribution of duration and magnitude of the upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$ respectively. These figures are included to further address the issue of overplotting by using a colour gradient to indicate the distribution of the WPREs across the range of ramp duration and ramp magnitude, instead of just making the points partially transparent.

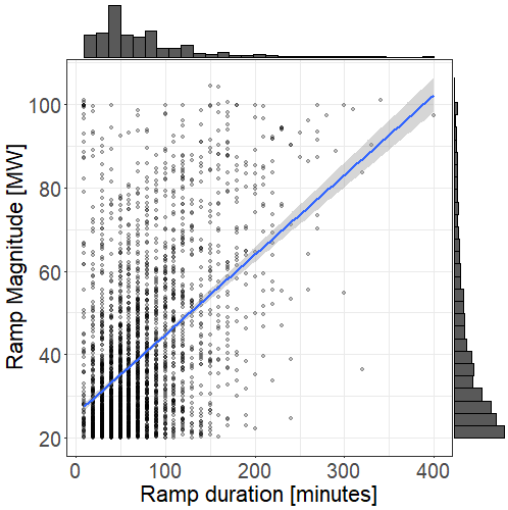


Fig. 94: Scatterplot of the joint distribution of the ramp duration and the ramp magnitude of the upward ramps detected by the multi-parameter segmentation algorithm and $\gamma = 0.01,\ \lambda = 5$, including the marginal distribution of the ramp duration and the ramp magnitude of the upward ramps.

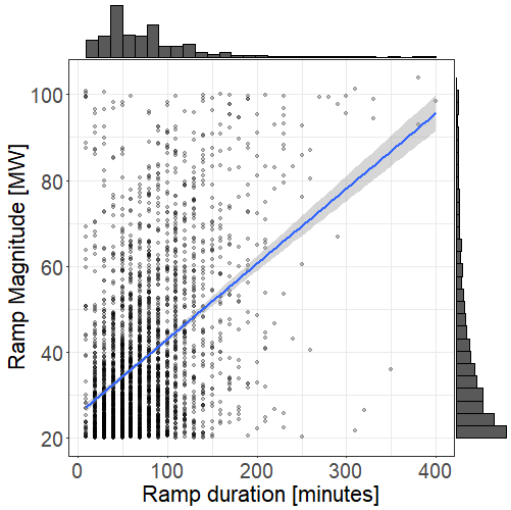


Fig. 95: Scatterplot of the joint distribution of the ramp duration and the ramp magnitude of the downward ramps detected by the multi-parameter segmentation algorithm and $\gamma = 0.01,\ \lambda = 5$, including the marginal distribution of the ramp duration and the ramp magnitude of the downward ramps.

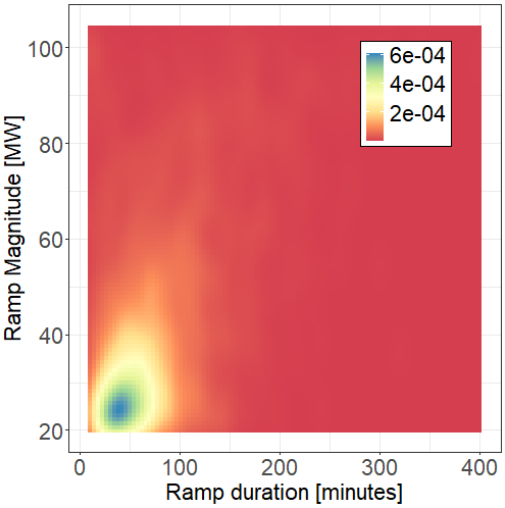


Fig. 96: Joint distribution of the ramp duration and the ramp magnitude of the upward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01$, $\lambda = 5$.

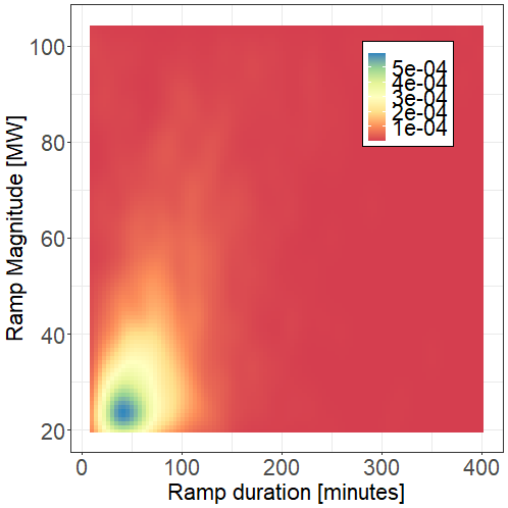


Fig. 97: Joint distribution of the ramp duration and the ramp magnitude of the downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01$, $\lambda = 5$.

Fig. 98 depicts the distribution of the ramp magnitude of the upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$. Fig. 98 is used to evaluate and compare the range and distribution of the ramp magnitude of the upward and downward ramps. Essentially, the relationship between the direction of the ramps and the magnitude of the ramps is analysed.

For the upward ramps, the minimum and maximum value observed for the ramp magnitude is 20.01 MW and 104.33 MW respectively, and the mean value of the ramp magnitude is 39.22 MW. Furthermore, the

value of the 1st Quartile of the ramp magnitude is 24.92 MW, therefore 25% of the detected upward ramps have a ramp magnitude between 20.01 MW and 24.92 MW. The median value of the ramp magnitude is 32.32 MW, therefore 50% of the detected upward ramps have a ramp magnitude less than or equal to 32.32 MW. The value of the 3rd Quartile is 47.72 MW, therefore 75% of the detected upward ramps have a ramp magnitude less than 47.72 MW. The Interquartile range is 22.797 MW which represents the range of the ramp magnitude for the middle 50% of the upward ramps, i.e. 50% of the upward ramps have a ramp magnitude between 24.92 MW and 47.72 MW. Therefore, only 25% of the upward ramps are expected to exceed 47.72 MW.

For the downward ramps, the minimum and maximum value observed for the ramp magnitude is 20.00 MW and 103.95 MW respectively, and the mean value of the ramp magnitude is 38.35 MW. Furthermore, the value of the 1st Quartile of the ramp magnitude is 24.46 MW, therefore 25% of the detected downward ramps have a ramp magnitude between 20.00 MW and 24.46 MW. The median value of the ramp magnitude is 32.05 MW, therefore 50% of the detected downward ramps have a ramp magnitude less than or equal to 32.05 MW. The value of the 3rd Quartile is 46.14 MW, therefore 75% of the detected downward ramps have a ramp magnitude less than 46.14 MW. The interquartile range is 21.68 MW and represents the range of the ramp magnitude for the middle 50% of the downward ramps, i.e. 50% of the downward ramps have a ramp magnitude between 24.46 MW and 46.14 MW. Therefore, only 25% of the upward ramps are expected to exceed 46.14 MW.

It is evident that the distribution of the ramp magnitude is similar for upward and downward ramps, specifically, the 1st Quartile, 3rd Quartile, median, maximum and minimum values of the ramp magnitude are approximately similar for the upward and downward ramps. The severity of the upward and downward ramps is, therefore, similar with regards to the capacity requirements imposed on the conventional generation units, and the distribution of the ramp magnitude is not related to the direction of the ramp. Both boxplots are right skewed, indicating that the magnitude of the upward and downward ramps are generally small, as mentioned above.

For both the upward and downward ramps, the ranges of the quartile groups vary. The bottom and top whiskers represent the ranges of the ramp magnitude for the bottom 25% and top 25% of the detected upward and downward ramps. The range of the ramp magnitude for the bottom 25% of both the detected upward and downward ramps is small, i.e. the lower whisker is short, indicating that the ramp magnitude of the bottom 25% of both the detected upward and downward ramps do not vary widely. However, the range of the ramp magnitude for the top 25% of both the detected upward and downward ramps is large, i.e. the upper whisker is long, indicating that the ramp magnitude of the top 25% of both the upward and downward ramps vary widely. Furthermore, upward ramps with a magnitude greater than approximately 78 MW is considered as outliers, and downward ramps with a magnitude greater than approximately 82 MW is considered as outliers.

Fig. 99 depicts the distribution of the ramp duration of the upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$. Fig. 99 is used to evaluate and compare the range and distribution of the ramp duration of the upward and downward ramps. Essentially, the relationship between the direction of the ramps and the duration of the ramps is analysed.

For the upward ramps, the minimum and maximum value observed for the ramp duration is 10 minutes and 400 minutes respectively, and the mean value is 70.98 minutes. Furthermore, the value of the 1st Quartile is 40 minutes, therefore 25% of the detected upward ramps have a ramp duration between 10

and 40 minutes. The median value is 60 minutes, therefore 50% of the detected upward ramps have a ramp duration less than or equal to 60 minutes. The value of the 3rd Quartile is 90 minutes, therefore 75% of the detected upward ramps have a ramp duration less than 90 minutes. The interquartile range is 50 minutes and represents the range of the ramp duration for the middle 50% of the upward ramps, i.e. 50% of the upward ramps have a ramp duration between 40 minutes and 90 minutes.

For the downward ramps, the minimum and maximum value observed for the ramp duration is 10 minutes and 400 minutes respectively, and the mean value is 73.31 minutes. Furthermore, the 1st Quartile, median and 3rd Quartile values of the ramp duration, as well as the value of the interquartile range, for the downward ramps correspond to that of the upward ramps.

It is evident from the above-mentioned information and Fig. 99 that the distribution of the ramp duration is similar for upward and downward ramps. Therefore, the severity of the upward and downward ramps is similar, with regards to the ramping requirements imposed on the conventional generation units. Additionally, the distribution of the ramp duration is not related the direction of the ramp. Both boxplots are right skewed, indicating that the duration of the upward and downward ramps are generally short, as mentioned above.

For both the upward and downward ramps, the ranges of the quartile groups vary. The bottom and top whiskers represent the ranges of the ramp duration for the bottom 25% and top 25% of the detected upward and downward ramps. The range of the ramp duration for the bottom 25% of both the detected upward and downward ramps is small, i.e. the lower whisker is short, indicating that ramp duration of the bottom 25% of both the detected upward and downward ramps do not vary widely. However, the range of the ramp duration for the top 25% of both the detected upward and downward ramps is large, i.e. a long upper whisker, indicating the ramp duration of the top 25% of both the upward and downward vary widely. Furthermore, upward and downward ramps with a duration greater than approximately 160 minutes are considered as outliers.

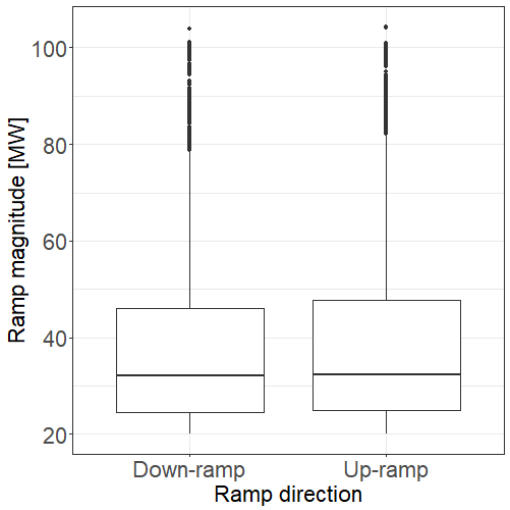


Fig. 98: Boxplot for the ramp magnitude of upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$.

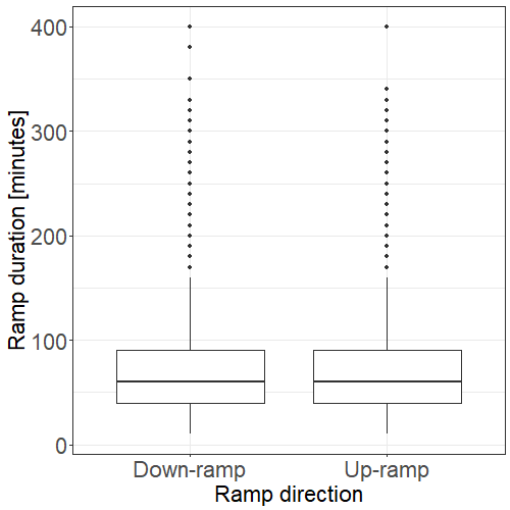


Fig. 99: Boxplot for the ramp duration of upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$.

The PDFs for the ramp magnitude and ramp duration of the upward and downward ramps detected by the multi-parameter segmentation algorithm and $\gamma = 0.01,\ \lambda = 5$ is shown in Fig. 49 to Fig. 52 in section 3.6.3, and a detailed discussion of the PDFs is also included.

Furthermore, a histogram representing the frequency distribution of the magnitude of the upward and downward ramps is shown in Fig. 100, i.e. Fig. 100 shows the number of upward ramps and downward ramps that has a magnitude that falls within a specific range. Fig. 101 shows a histogram representing the frequency distribution of the duration of the upward and downward ramps, i.e. Fig. 101 shows the number of upward ramps and downward ramps that has a duration that falls within a specific range. The multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$ detected 3118 upward ramps and 3188 downward ramps.

The frequency distribution of the ramp magnitude is approximately similar for the upward and downward ramps, as previously determined by the boxplots in Fig. 98. For smaller ramp magnitude values, specifically if the ramp magnitude is in the range of 20 to 50 MW, the number of downward ramps exceeds the number of upward ramps. Generally, for larger ramp magnitude values, specifically if the ramp magnitude exceeds 70 MW, the number of upward ramps exceeds the number of downward ramps. Therefore, it is slightly more likely that large upward ramps will occur than large downward ramps.

The frequency distribution of the ramp duration is approximately similar for the upward and downward ramps, as previously determined by the boxplots in Fig. 99. For smaller ramp duration values, specifically if the ramp duration is in the range of 0 to 40 minutes, the number of upward ramps exceeds the number of downward ramps. If the ramp duration is in the range of 40 to 140 minutes, the number of downward ramps exceeds the number of upward ramps. If the ramp duration exceeds 140 minutes, the number of upward ramps exceeds the number of downward ramps. Therefore, it is slightly more likely that short upward ramps will occur than short downward ramps.

Fig. 102 depicts the distribution for the number of ramps observed in a day, irrespective of the direction, as well as the distribution for the number of upward and downward ramps observed in a day. The mean number of ramps observed in a day is 5.759 ramps. The maximum and minimum number of ramps observed in a day is 31 ramps and 0 ramps, respectively. The 1st Quartile value for the number of ramps in a day is 3, therefore 25% of the days during the 3-year period have 3 ramps or less. The median value for the number of ramps per day is 6, therefore 50% of the days during the 3-year period have 6 ramps or less. The 3rd Quartile value for the number of ramps per day is 8, therefore 75% of the days during the 3-year period have 8 ramps or less.

Furthermore, the distribution for the number of upward ramps per day is similar to the distribution for the number of downward ramps per day. The minimum number of upward ramps and downward ramps observed per day is 0, and the maximum number of upward ramps and downward ramps observed per day is 15 and 16 ramps, respectively. The 1st Quartile, median and 3rd Quartile value for the number of upward ramps, as well as the number of downward ramps, occurring in a day is 1, 3 and 4 ramps, for both cases, respectively. Additionally, 2.847 upward ramps occur on average per day and 2.911 downward ramps occur on average per day.

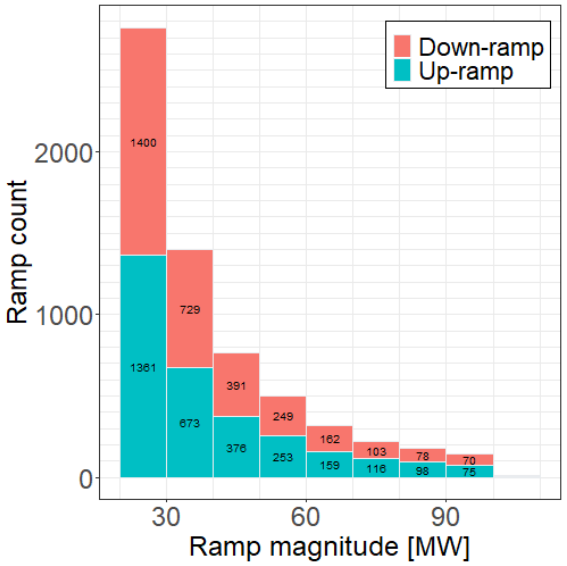


Fig. 100: The frequency distribution of the magnitude of the upward and downward ramps detected using the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$.

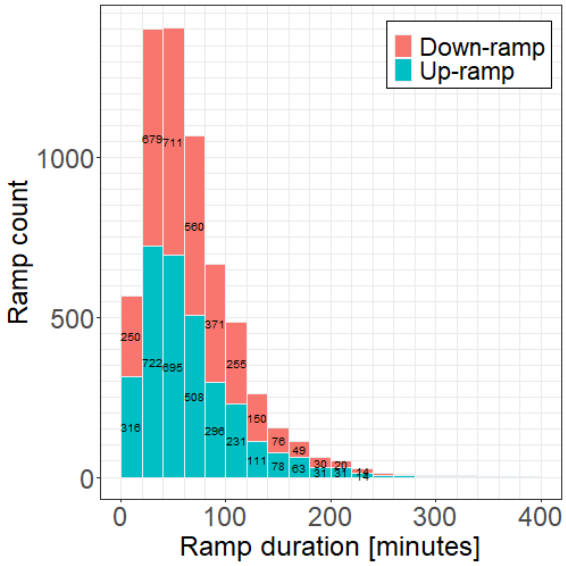


Fig. 101: The frequency distribution of the duration of the upward and downward ramps detected using the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$.

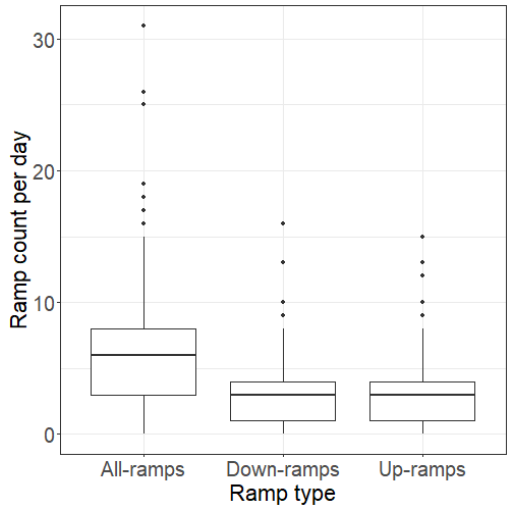


Fig. 102: The distribution for the number of ramps observed per day, irrespective of the direction, as well as the distribution for the number of upward and downward ramps observed per day.

Fig. 103 to Fig. 110 shows the seasonality in the ramps detected via the multi-parameter segmentation algorithm with $\gamma = 0.01,\ \lambda = 5$. Fig. 103 shows the hourly distribution of the ramp magnitude of the detected upward and downward ramps Fig. 104 shows the up-ramp count and down-ramp count for each hour of the day over a period of 3 years. Fig. 105 and Fig. 106 shows the hourly patterns of the up-ramp count and down-ramp count respectively, i.e. Fig. 105 and Fig. 106 shows the number of up-ramps and down-ramps, respectively, for each hour of the day per year for three years. Fig. 107 shows the monthly distribution of the ramp magnitude of the detected upward and downward ramps. Fig. 108 shows the up-

ramp count and down-ramp count for each month of the year over a period of 3 years. Fig. 109 and Fig. 110 shows the monthly patterns of the number of up-ramps and down-ramps, respectively. Specifically, Fig. 109 and Fig. 110 shows the number of up-ramps and down-ramps respectively, for each month of the year, per year.

From Fig. 103, it is recognised that the distribution of the ramp magnitude of the upward and downward ramps is dependent on the hour of the day, i.e. different distributions are observed for the magnitude of the upward and downward ramps at each of the hour of the day. The mean ramp magnitude observed per hour for upward ramps vary between 34.02 MW and 43.84 MW, and the mean ramp magnitude observed per hour for downward ramps vary between 34.03 MW and 42.63 MW. The maximum ramp magnitude observed per hour for upward ramps vary between 80.68 MW and 104.33 MW, and the maximum ramp magnitude observed per hour for downward ramps vary between 84.43 MW and 103.95 MW. The minimum ramp magnitude for upward ramps observed per hour vary between 20.01 MW and 20.94 MW, and the minimum ramp magnitude for downward ramps observed per hour vary between 20.01 MW and 20.4 MW. The median ramp magnitude for upward ramps observed per hour vary between 29.37 MW and 37.38 MW, and the median ramp magnitude for downward ramps observed per hour vary between 28.89 MW and 36.18 MW. The 3rd Quartile value of ramp magnitude for upward ramps per hour vary between 38.98 MW and 55.72 MW and the 3rd Quartile value of ramp magnitude for downward ramps observed per hour vary between 38.01 MW and 56.83 MW. The interquartile range of ramp magnitude for upward ramps observed per hour vary between 14.73 MW and 28.96 MW, and the interquartile range ramp magnitude for downward ramps observed per hour vary between 14.25 MW and 31.3 MW. Therefore, it is evident that the hourly distributions for the ramp magnitude of the upward and downward ramps varies widely.

Furthermore, the interquartile range for the magnitude of upward ramps is the greater than 25 MW at 03:00 and 04:00, as well as at 09:00 to 12:00, thus the magnitude of the middle 50% of the upward ramps varies most widely during these hours. Additionally, the 3rd Quartile value for the magnitude of the upward ramps is greater than 50 MW at 03:00, 04:00, and 09:00 to 12:00, as well as at 23:00, therefore larger upward ramps are more likely to occur during these hours of the day, compared to the remaining hours. Conversely, the interquartile range for the magnitude of downward ramps is the greater than 25 MW at 10:00, as well as at 19:00 to 21:00, thus the magnitude of the middle 50% of the downward ramps varies most widely during these hours. Additionally, at 10:00, as well as at 19:00 to 21:00, a 3rd Quartile value higher than 50 MW is observed. Therefore, larger downward ramps are more likely to occur during these hours of the day, compared to the remaining hours.

From Fig. 104 to Fig. 106, it is evident that the number of upward and downward ramps are dependent on the hour of the day, and upward and downward ramps are more likely to occur during different times of the day. It is shown that upward ramps occur more frequently around midday and the afternoon, while downward ramps occur more frequently during the evening as well as at night-time, which may be the result of daily weather patterns.

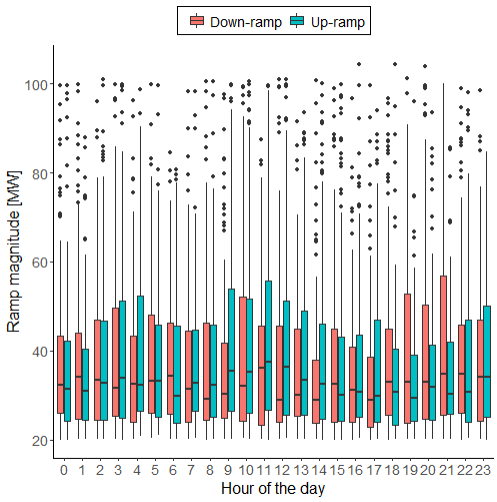


Fig. 103: Hourly distribution for the ramp magnitude of the upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$.

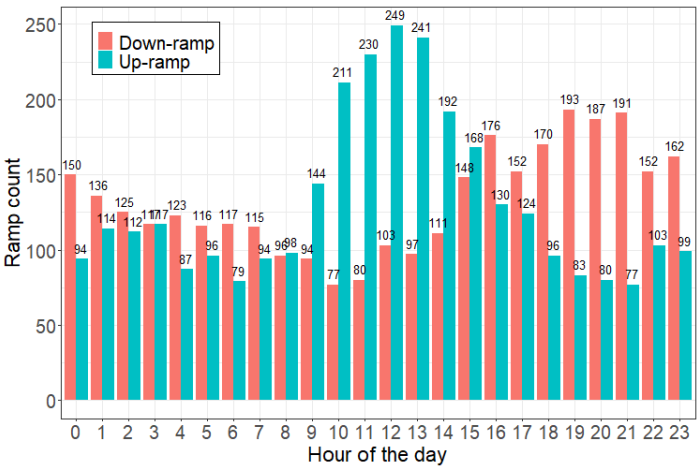


Fig. 104: Number of upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$ per hour.

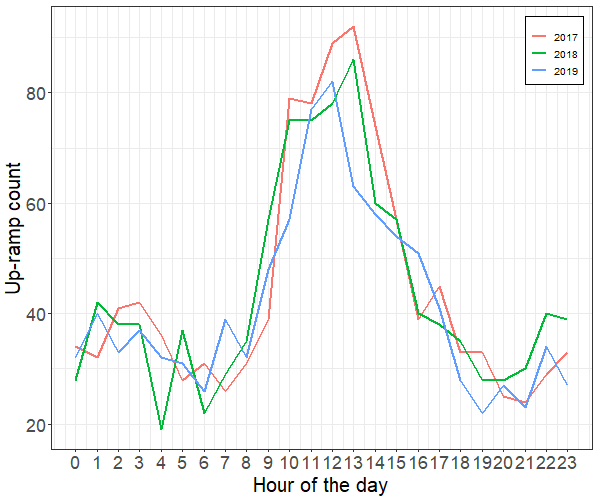


Fig. 105: Number of upward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$ per hour for each year.

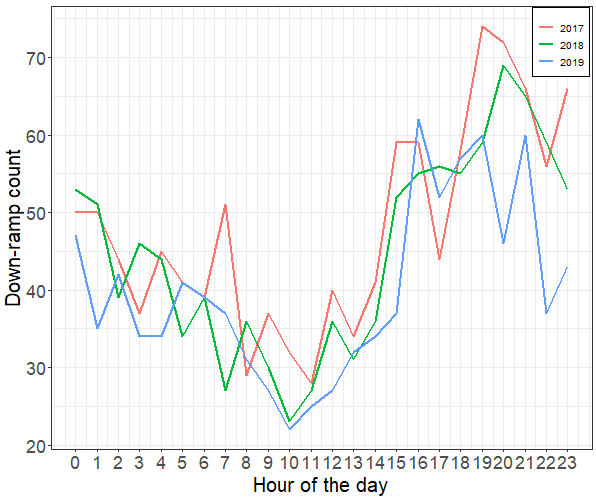


Fig. 106: Number of downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01$, $\lambda = 5$ per hour for each year.

From Fig. 107, it is evident that during a great portion of the months of the year the monthly distribution for the ramp magnitude of upward ramps is approximately similar to the monthly distribution for the ramp magnitude of downward ramps. However, during some months the monthly distribution for the ramp magnitude of the upward ramps varies more widely than the monthly distribution of the downward ramps. The mean ramp magnitude observed per month for upward ramps vary between 36.94 MW and 42.62 MW, and the mean ramp magnitude observed per month for downward ramps vary between 36.33 MW and 41.10 MW. The maximum ramp magnitude observed per month for upward ramps vary between 96.96 MW and 104.33 MW, and the maximum ramp magnitude observed per month for downward ramps vary between 96.64 MW and 103.95 MW. The minimum ramp magnitude for upward ramps observed per

hour vary between 20.01 MW and 20.3 MW, and the minimum ramp magnitude for downward ramps observed per hour vary between 20.01 MW and 20.15 MW. The median ramp magnitude for upward ramps observed per month vary between 30.56 MW and 35.27 MW, and the median ramp magnitude for downward ramps observed per month vary between 29.67 MW and 33.75 MW. The interquartile range of ramp magnitude for upward ramps observed per month vary between 19.16 MW and 27.53 MW, and the interquartile range of ramp magnitude for downward ramps observed per month vary between 19.16 MW and 27.51 MW. The 3rd Quartile value of ramp magnitude for upward ramps observed per month vary between 43.55 MW and 53.19 MW, and the 3rd Quartile value of ramp magnitude for downward ramps observed per month vary between 43.01 MW and 51.49 MW.

The interquartile range for the magnitude of upward ramps is the greater than 24 MW during February, April, July and November, thus the magnitude of the middle 50% of the upward ramps varies most widely during February, April, July and November. Additionally, the 3rd Quartile value for the magnitude of the upward ramps is greater than 50 MW during February, April and November, therefore larger upward ramps are more likely to occur during February, April and November, compared to the remaining months of the year. Conversely, the interquartile range for the magnitude of the downward ramps is the greater than 24 MW during February and November, thus the magnitude of the middle 50% of the downward ramps varies most widely during February and November. Additionally, during February, a 3rd Quartile value higher than 50 MW is observed. Therefore, larger downward ramps are more likely to occur during February, compared to the remaining months of the year.

From Fig. 108 to Fig. 110, it is evident that the number of up and down ramps are dependent on the month of the year, and a similar monthly trend is observed for the upward and downward ramps. It is shown that both upward and downward ramps occur more frequently during summer and the spring, and less frequently during autumn and winter, which may be the result of seasonal weather patterns.

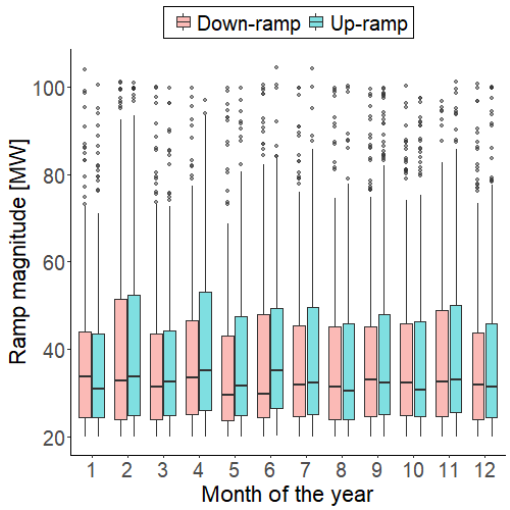


Fig. 107: Monthly distribution for the ramp magnitude of upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$.

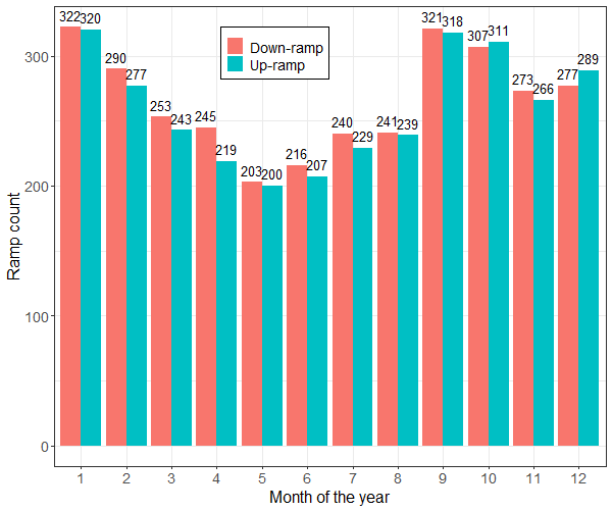


Fig. 108: Number of upward and downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$ per hour .

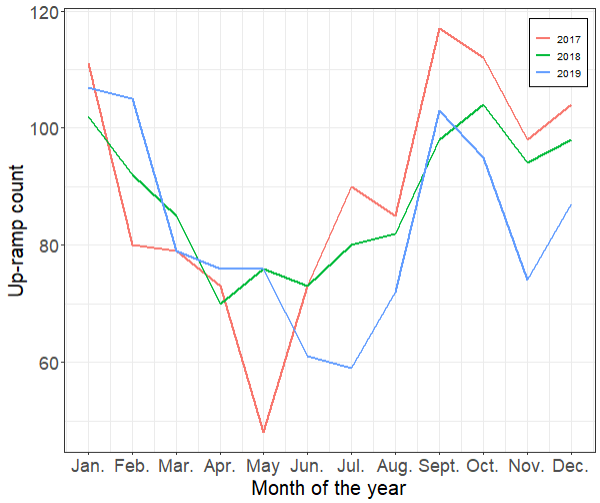


Fig. 109: Number of upward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01$, $\lambda = 5$ per month for each year.

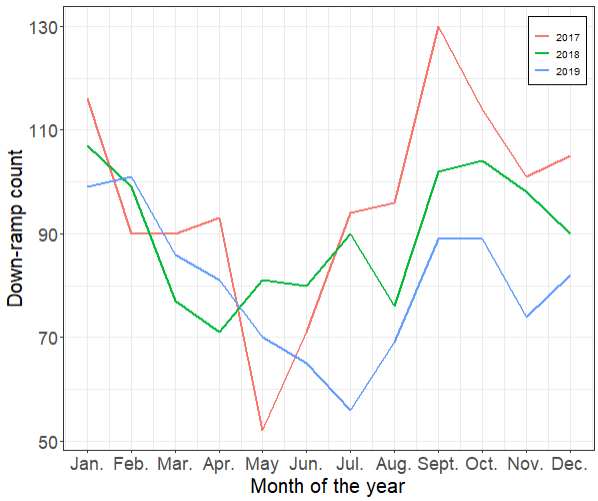


Fig. 110: Number of downward ramps detected by the multi-parameter segmentation algorithm for $\gamma = 0.01$, $\lambda = 5$ per month for each year.

Fig. 111 shows the probability density function of the interarrival times of ramps. The interarrival time is defined as the difference in time between the start of two ramps. This information is valuable to system operators as it provides information about the ramping requirements of conventional generation sources. Therefore, the interarrival time was calculated between (i) all consecutive ramps irrespective of the direction, (ii) all downward ramps, (iii) all upward ramps, (iv) all downward ramps and its consecutive upward ramps, as well as (v) all upward ramps and it consecutive downward ramps. It is important to note that for (iv), the interarrival time was only calculated between the upward and downward ramps for the upward ramps that are directly followed by a downward ramp, and for (v) the interarrival time was only calculated between downward and upward ramps for downward ramps which are directly followed by an upward ramp.

It is evident that all the probability density functions are right skewed which indicate that the interarrival times for (i)-(v) are generally short. The mean interarrival time observed for (i) to (v) is 249 min, 492.5 min, 503.5 min, 353.6 min and 203.8 min, respectively. The minimum interarrival time observed for (i) to (v) is 10 min, 20 min, 20 min, 10 min and 10 min, respectively. The maximum interarrival time observed for (i) to (v) is 270 min, 5470 min, 6920 min, 5430 min and 2750 min, respectively. The Interquartile range for the interarrival time is between 70 and 270 min for (i), 120 and 680 min for (ii), 130 and 690 min for (iii) , 70 and 430 min for (iv), as well as 50 and 250 min for (v).

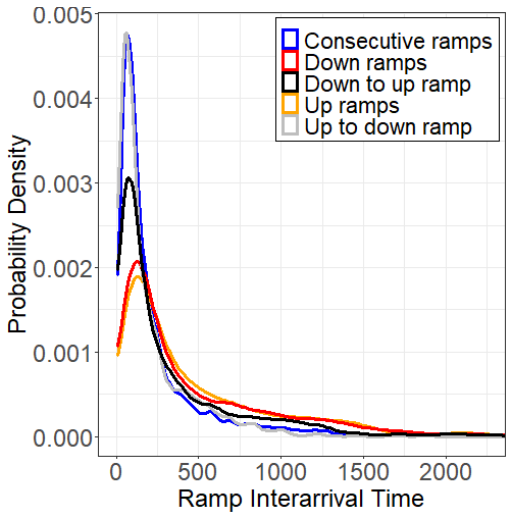


Fig. 111: Interarrival time for (i) all ramps, (ii) downward ramps, (iii) upward ramps, (iv) consecutive downward and upward ramps, as well as (v) consecutive upward and downward ramps.

5 Cluster analysis

5.1 Overview

This chapter presents and investigates the application of clustering analysis to the ramp detection results obtained via the multi-parameter segmentation algorithm for $\gamma = 0.1, \lambda = 5$. Specifically, clustering is applied to a dataset containing the key ramp features of the detected upward ramps, as well as to a dataset containing the key ramp features of the detected downward ramps. Each observation in the respective datasets represents a Wind Power Ramp Event (WPRE) which is represented by two variables namely the ramp duration and ramp magnitude. These observations can be represented as coordinates in a 2-dimensional dataspace. Cluster analysis is used to explore the data in order to identify underlying patterns or structures in the data and to get insights into the distribution of the dataset of interest, with the aim to characterise the wind energy facility site in terms of ramping mode.

A diverse range of clustering algorithms, distance measures and linkage criteria are investigated in this study to determine the optimal clustering procedure for the datasets of interest. The clustering methods considered in the study to classify the observations of the datasets into multiple groups include k-means, PAM, agglomerative clustering, Divisive Analysis (DIANA) and c-means. The distance measures considered to calculate the distance between a pair of observations, in order to decide which observations should be grouped together, include the Euclidean, Manhattan and maximum distance measures. For the agglomerative clustering algorithm, 4 different linkage criteria, i.e. complete, single, average and Ward, are considered to calculate the distance between non-singleton clusters. Furthermore, since the ramp magnitude and ramp duration are measured in different units, it is necessary to standardise the data before computing the distance measures to make these variables comparable. The min-max normalisation technique listed in (65) is employed to standardise the scale of the variables.

This chapter is organised as follows. Firstly, the determination of the optimal number of clusters for each clustering methodology is presented, followed by the results obtained for each clustering methodology. Next, the analysis of the validity of the clusters produced by each clustering methodology, as well as a comparison of these results is provided. Finally, summary statistics are presented for the optimal clustering results so to characterise the wind energy facility site.

5.2 Optimal number of clusters

The optimal number of clusters are determined for the k-means, PAM, agglomerative, DIANA and c-means clustering methodologies for the different distance measures and linkage criteria, where applicable. The k-means and PAM clustering algorithms require the user to specify the number of clusters, k, to be generated. Hierarchical clustering algorithms, i.e. agglomerative and DIANA, do not require the number of clusters to be pre-specified. It is, however, possible to cut the dendrogram, i.e. a tree-like representation of the results of the hierarchical clustering, at a specific height to yield suitable clusters. Furthermore, it is possible to obtain the crisp clustering results of the c-means clustering algorithm.

Two popular methods are used to determine the optimal number of clusters for the k-means, PAM, agglomerative, DIANA and c-means clustering algorithms, namely the elbow method and the silhouette method. The elbow method calculates and plots the total within cluster sum of squares for a range of k values. The optimal number of clusters is the k value which corresponds to a bend in the curve. The bend signifies the point where a further increase in k will not significantly change the within cluster sum of

squares. The silhouette method calculates and plots the average silhouette value for a range of k values. The optimal number of clusters is the k value which maximises the average silhouette value.

The results for the elbow method and silhouette method for each clustering methodology as obtained for each dataset, i.e. the dataset for the upward and downward ramps respectively, are presented in Appendix B**.** Table 20 summarises the optimal number of clusters for each combination of clustering method, distance measure and linkage criteria, where applicable, as obtained by the silhouette method, since the results of the elbow method is ambiguous.

Table 20: Optimal number of clusters for each clustering method.

Clustering algorithm	**Distance metric**	**Linkage criterion**	**Optimal number of clusters for upward ramps**	**Optimal number of clusters for downward ramps**
K-means	Euclidean	NA	2	2
	Manhattan	NA	2	2
	Maximum	NA	2	2
PAM	Euclidean	NA	2	2
	Manhattan	NA	2	2
	Maximum	NA	2	2
DIANA	Euclidean	NA	2	2
	Manhattan	NA	3	2
	Maximum	NA	2	2
C-means	Euclidean	NA	2	2
	Manhattan	NA	2	2
	Maximum	NA	2	2
Agglomerative clustering	Euclidean	Average	3	2
		Complete	3	2
		Single	2	7
		Ward	3	3
	Manhattan	Average	2	2
		Complete	2	6
		Single	2	9
		Ward	2	2
	Maximum	Average	3	2
		Complete	3	5
		Single	2	7
		Ward	2	3

5.3 Clustering results

5.3.1 Overview

This section presents the different clustering results as obtained for the detected upward and downward ramps by varying all combinations of the clustering methods, the distance measures and the linkage criteria, where applicable, considered in the study. The clustering results is presented by visualising the observations of the relevant dataset, i.e. the upward or downward ramps, via a scatterplot and colouring each observation according to its cluster assignment. A visualisation of the silhouette information corresponding to the different clustering results is also provided. The results of the agglomerative algorithm are also represented using dendrograms.

5.3.2 Results for k-means clustering

5.3.2.1 Upward ramps

The non-overlapping clusters obtained for the detected upward ramps with the k-means algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 112, Fig. 114 and Fig. 116, respectively. The silhouette information according to the clustering of the upward ramps with the k-means algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 113, Fig. 115 and Fig. 117, respectively.

Fig. 112 shows that clustering the upward ramps with the k-means algorithm using the Euclidean distance metric into 2 groups resulted in 2 cluster sizes of 728 and 2390, respectively, and it is recognised from Fig. 113 that 54 observations yielded negative silhouette widths. Fig. 114 shows that clustering the upward ramps with the k-means algorithm using the Manhattan distance metric into 2 groups resulted in 2 cluster sizes of 692 and 2426, respectively, and it is recognised from Fig. 115 that 46 observations yielded negative silhouette widths. Fig. 116 shows that clustering the upward ramps with the k-means algorithm using the maximum distance metric into 2 groups resulted in 2 cluster sizes of 755 and 2363, respectively, and it is recognised from Fig. 117 that 60 observations yielded negative silhouette widths.

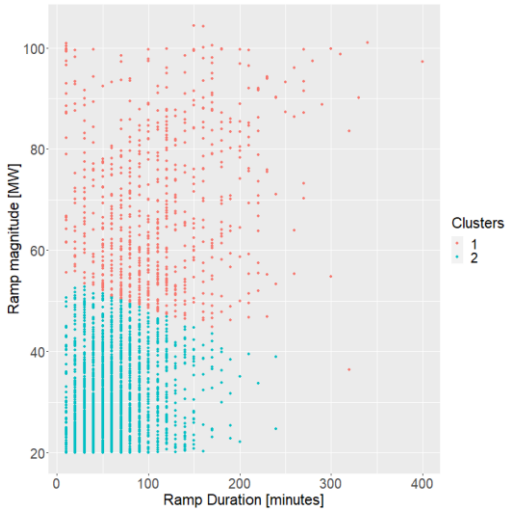


Fig. 112: Clusters obtained for the upward ramps with the k-means algorithm for the Euclidean distance metric.

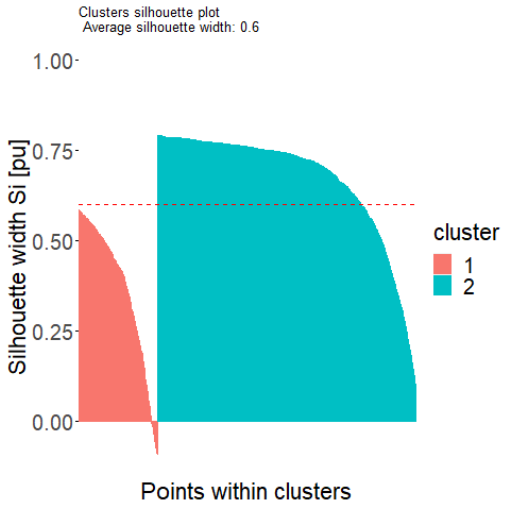


Fig. 113: Silhouette information according to the clustering of the upward ramps with the k-means algorithm for the Euclidean distance metric.

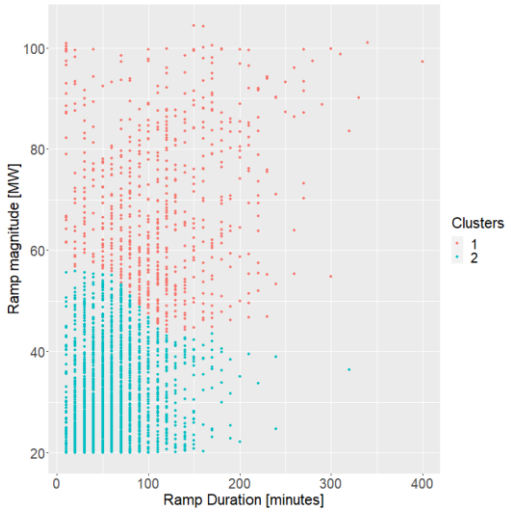


Fig. 114: Clusters obtained for the upward ramps with the k-means algorithm for the Manhattan distance metric.

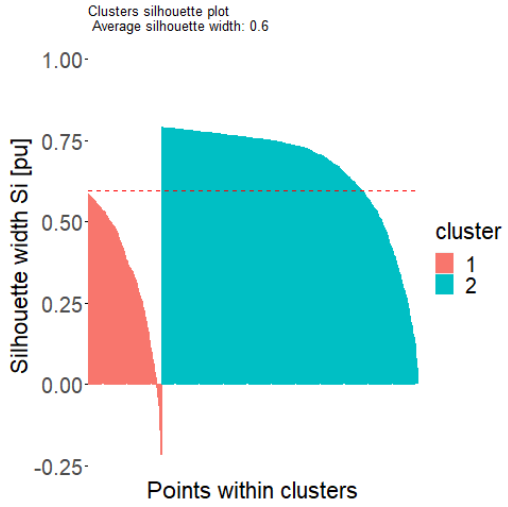


Fig. 115: Silhouette information according to the clustering of the upward ramps with the k-means algorithm for the Manhattan distance metric.

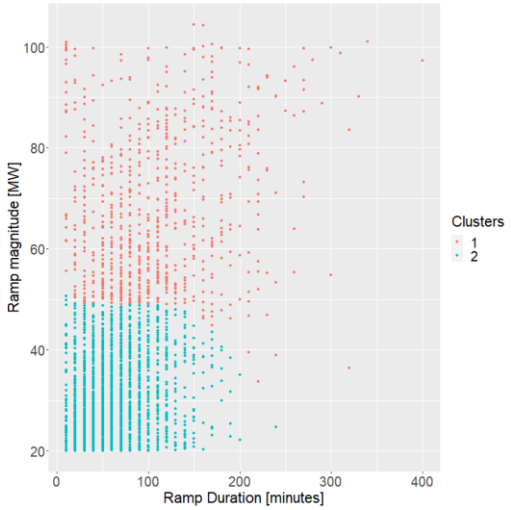


Fig. 116: Clusters obtained for the upward ramps with the k-means algorithm for the maximum distance metric.

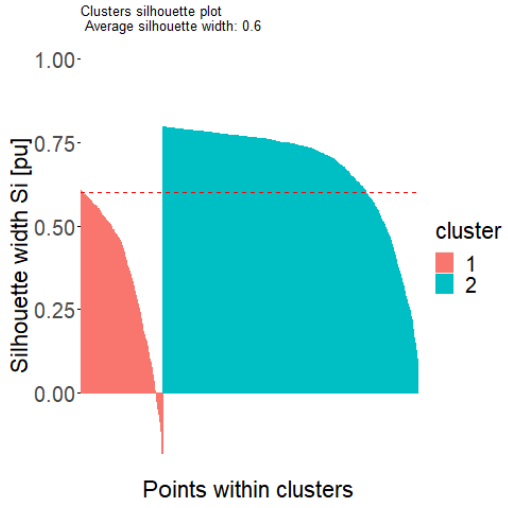


Fig. 117: Silhouette information according to the clustering of the upward ramps with the k-means algorithm for the maximum distance metric.

5.3.2.2 Downward ramps

The non-overlapping clusters obtained for the detected downward ramps with the k-means algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 118, Fig. 120 and Fig. 122, respectively. The silhouette information according to the clustering of the downward ramps with the k-means algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 119, Fig. 121 and Fig. 123, respectively.

Fig. 118 shows that clustering the downward ramps with the k-means algorithm using the Euclidean distance metric into 2 groups resulted in 2 cluster sizes of 2497 and 691, respectively, and it is recognised from Fig. 119 that 33 observations yielded negative silhouette widths. Fig. 120 shows that clustering the

downward ramps with the k-means algorithm using the Manhattan distance metric into 2 groups resulted in 2 cluster sizes of 2517 and 671, respectively, and it is recognised from Fig. 121 that 41 observations yielded negative silhouette widths. Fig. 122 shows that clustering the downward ramps with the k-means algorithm using the maximum distance metric into 2 groups resulted in 2 cluster sizes of 2439 and 749, respectively, and it is recognised from Fig. 123 that 60 observations yielded negative silhouette widths.

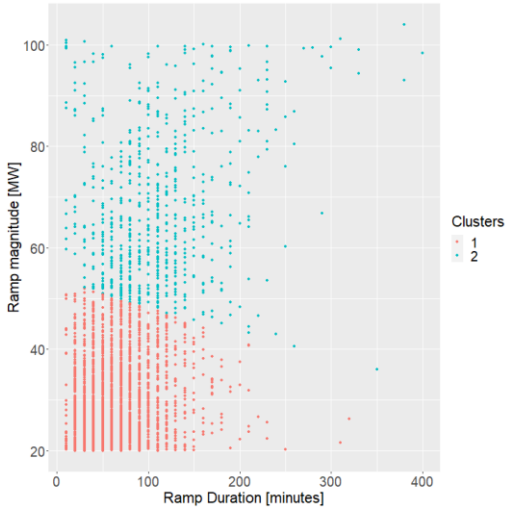


Fig. 118: Clusters obtained for the downward ramps with the k-means algorithm for the Euclidean distance metric.

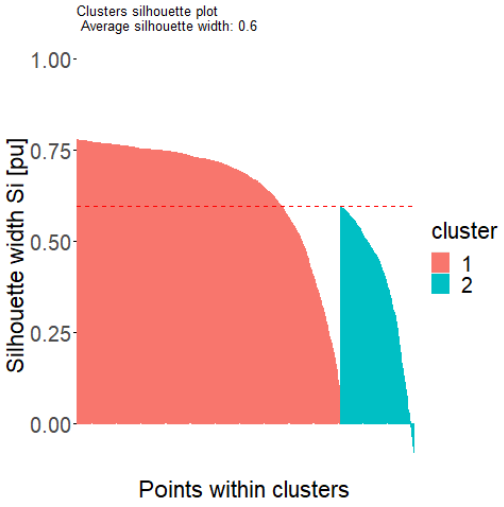


Fig. 119: Silhouette information according to the clustering of the downward ramps with the k-means algorithm for the Euclidean distance metric.

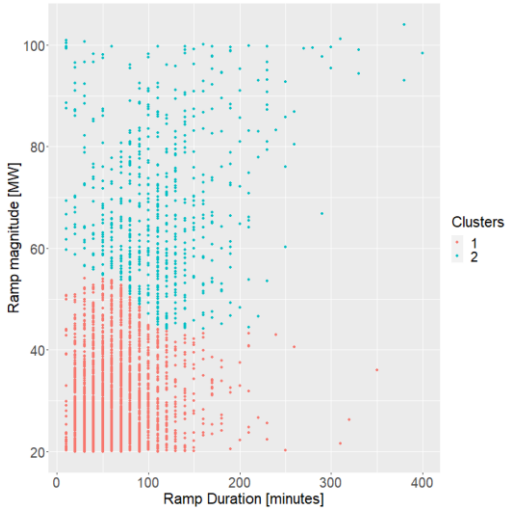


Fig. 120: Clusters obtained for the downward ramps with the k-means algorithm for the Manhattan distance metric.

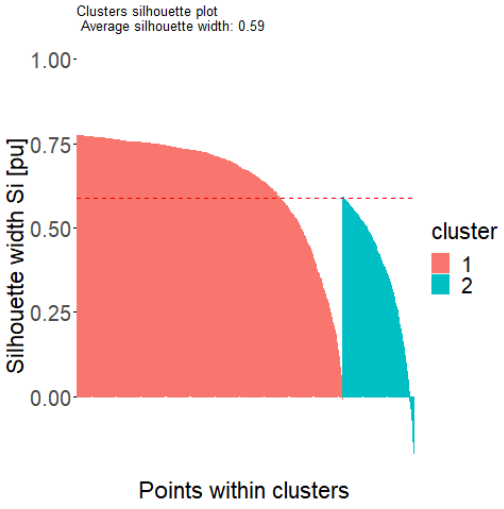


Fig. 121: Silhouette information according to the clustering of the downward ramps with the k-means algorithm for the Manhattan distance metric.

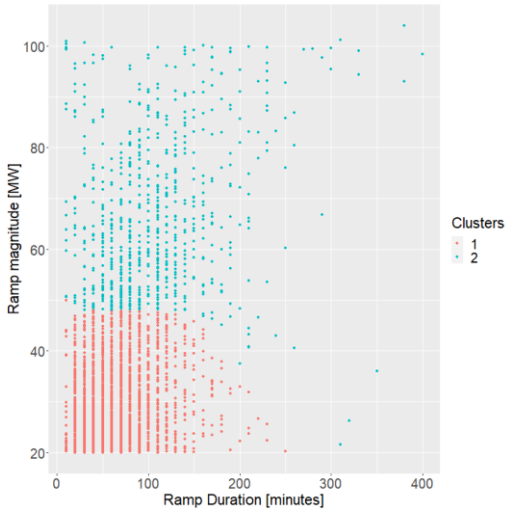


Fig. 122: Clusters obtained for the downward ramps with the k-means algorithm for the maximum distance metric.

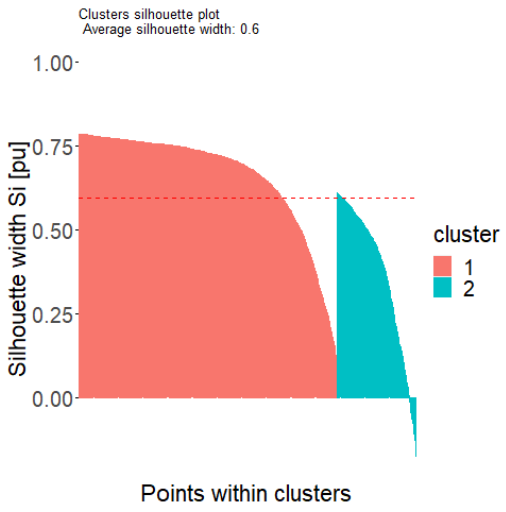


Fig. 123: Silhouette information according to the clustering of the downward ramps with the k-means algorithm for the maximum distance metric.

5.3.3 Results for partitioning around medoids clustering

5.3.3.1 Upward ramps

The non-overlapping clusters obtained for the detected upward ramps with the Partitioning Around Medoids (PAM) algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 124, Fig. 126 and Fig. 128, respectively. The silhouette information according to the clustering of the upward ramps with the PAM algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 125, Fig. 127 and Fig. 129, respectively.

Fig. 124 shows that clustering the upward ramps with the PAM algorithm using the Euclidean distance metric into 2 groups resulted in 2 cluster sizes of 2195 and 923, respectively, and it is recognised from Fig. 125 that 167 observations yielded negative silhouette widths. Fig. 126 shows that clustering the upward ramps with the PAM algorithm using the Manhattan distance metric into 2 groups resulted in 2 cluster sizes of 2201 and 917, respectively, and it is recognised from Fig. 127 that 169 observations yielded negative silhouette widths. Fig. 128 shows that clustering the upward ramps with the PAM algorithm using the maximum distance metric into 2 groups resulted in 2 cluster sizes of 2186 and 932, respectively, and it is recognised from Fig. 129 that 175 observations yielded negative silhouette widths.

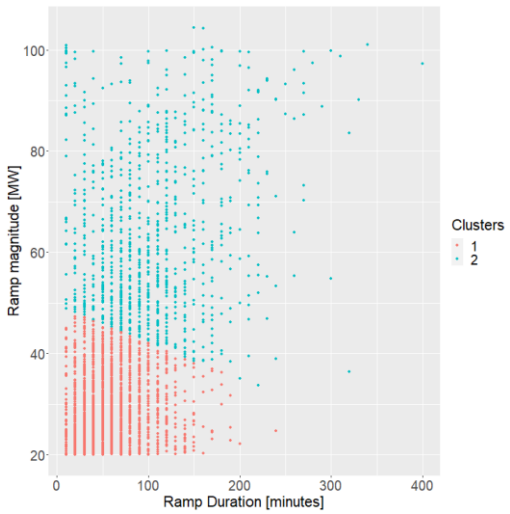


Fig. 124: Clusters obtained for the upward ramps with the PAM algorithm for the Euclidean distance metric.

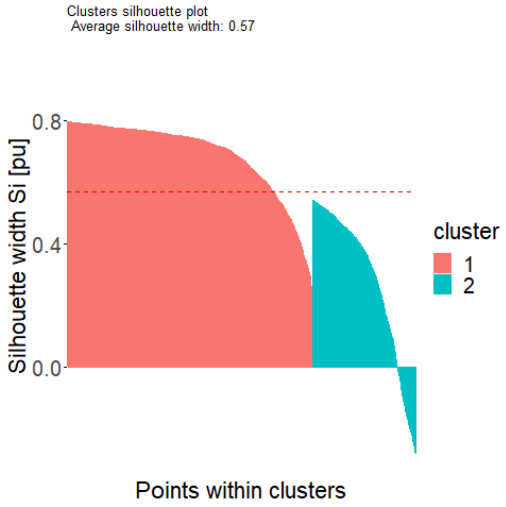


Fig. 125: Silhouette information according to the clustering of the upward ramps with the PAM algorithm for the Euclidean distance metric.

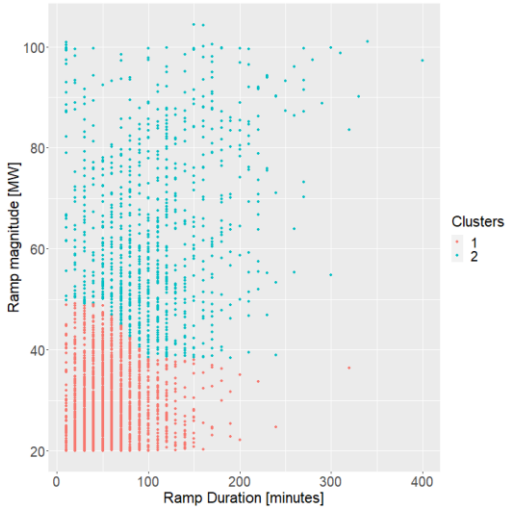


Fig. 126: Clusters obtained for the upward ramps with the PAM algorithm for the Manhattan distance metric.

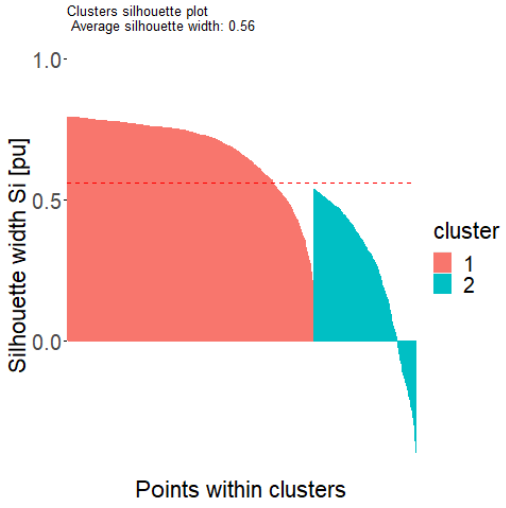


Fig. 127: Silhouette information according to the clustering of the upward ramps with the PAM algorithm for the Manhattan distance metric.

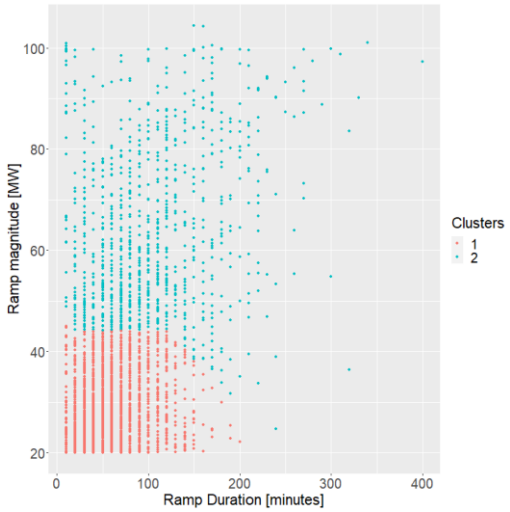


Fig. 128: Clusters obtained for the upward ramps with the PAM algorithm for the maximum distance metric.

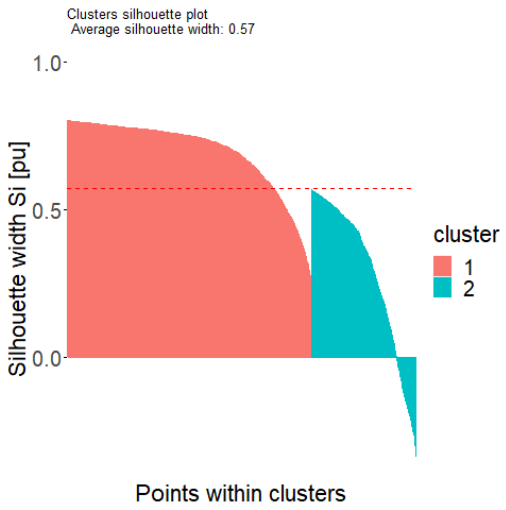


Fig. 129: Silhouette information according to the clustering of the upward ramps with the PAM algorithm for the maximum distance metric.

5.3.3.2 Downward ramps

The non-overlapping clusters obtained for the detected downward ramps with the PAM algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 130, Fig. 132 and Fig. 134, respectively. The silhouette information according to the clustering of the downward ramps with the PAM algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 131, Fig. 133 and Fig. 135, respectively.

Fig. 130 shows that clustering the downward ramps with the PAM algorithm using the Euclidean distance metric into 2 groups resulted in 2 cluster sizes of 2280 and 908, respectively, and it is recognised from Fig. 131 that 158 observations yielded negative silhouette widths. Fig. 132 shows that clustering the downward ramps with the PAM algorithm using the Manhattan distance metric into 2 groups resulted in 2 cluster sizes of 2259 and 929, respectively, and it is recognised from Fig. 133 that 172 observations yielded negative silhouette widths. Fig. 134 shows that clustering the downward ramps with the PAM algorithm using the maximum distance metric into 2 groups resulted in 2 cluster sizes of 2248 and 940, respectively, and it is recognised from Fig. 135 that 184 observations yielded negative silhouette widths.

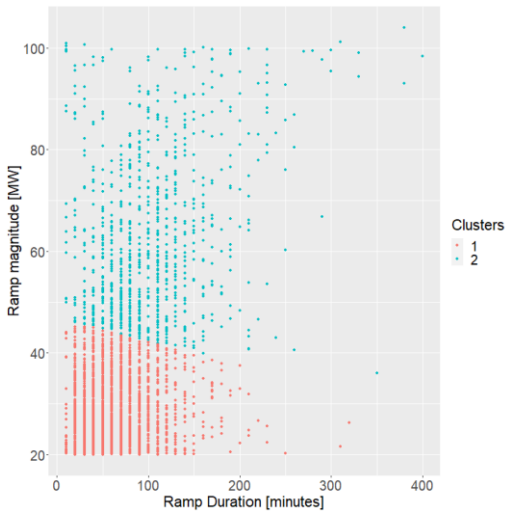


Fig. 130: Clusters obtained for the downward ramps with the PAM algorithm for the Euclidean distance metric.

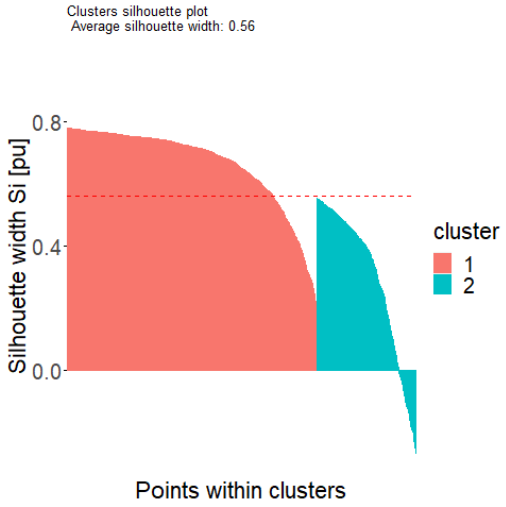


Fig. 131: Silhouette information according to the clustering of the downward ramps with the PAM algorithm for the Euclidean distance metric.

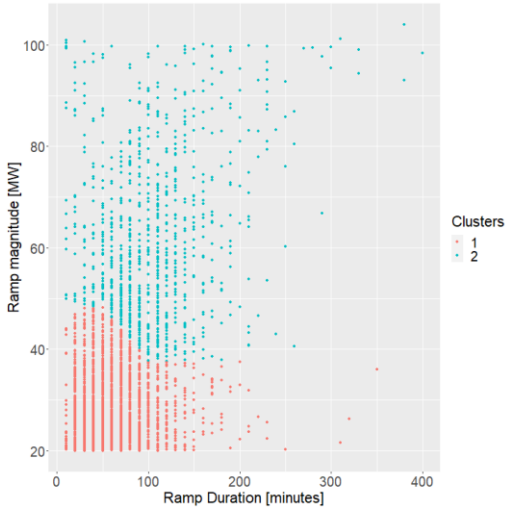


Fig. 132: Clusters obtained for the downward ramps with the PAM algorithm for the Manhattan distance metric.

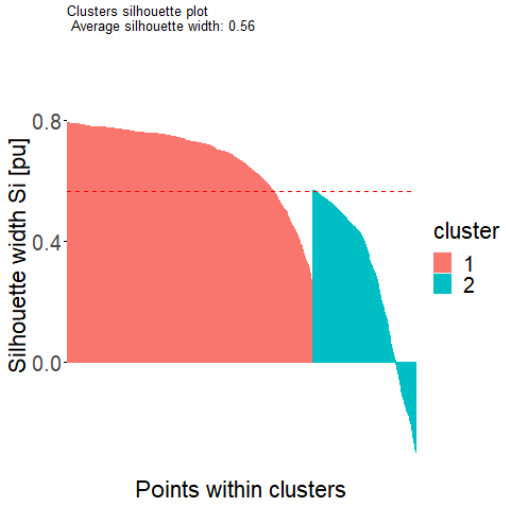


Fig. 133: Silhouette information according to the clustering of the downward ramps with the PAM algorithm for the Manhattan distance metric.

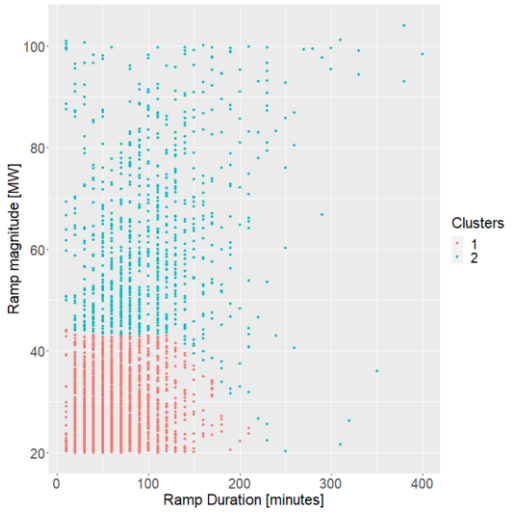


Fig. 134: Clusters obtained for the downward ramps with the PAM algorithm for the maximum distance metric.

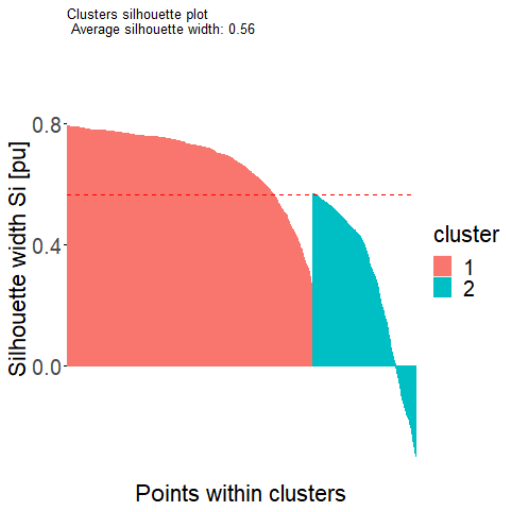


Fig. 135: Silhouette information according to the clustering of the downward ramps with the PAM algorithm for the maximum distance metric.

5.3.4 Results for agglomerative clustering

5.3.4.1 Upward Ramps

Fig. 284 to Fig. 287 in Appendix C show a dendrogram of the results of the agglomerative algorithm with the Euclidean distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the upward ramps. A dendrogram is a tree-like structure that illustrates the order in which the clusters were merged during each step.

The non-overlapping clusters obtained for the detected upward ramps with the agglomerative algorithm for the Euclidean distance metric and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 136, Fig. 138, Fig. 140 and Fig. 142. The silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Euclidean distance metric and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 137, Fig. 139, Fig. 141 and Fig. 143.

Fig. 136 shows that clustering the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the average linkage criterion and cutting the resulting dendrogram to yield 3 groups resulted in 3 cluster sizes of 2750, 366 and 2, respectively, and it is recognised from Fig. 137 that 150 observations yielded negative silhouette widths. Fig. 138 shows that clustering the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the complete linkage criterion and cutting the resulting dendrogram to yield 3 groups resulted in 3 cluster sizes of 2715, 362 and 41, respectively, and it is recognised from Fig. 139 that 157 observations yielded negative silhouette widths. Fig. 140 shows that clustering the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the single linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 3117 and 1, respectively, and it is recognised from Fig. 141 that 93 observations yielded negative silhouette widths. Fig. 142 shows that clustering the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the Ward linkage criterion and

cutting the resulting dendrogram to yield 3 groups resulted in 3 cluster sizes of 1840, 839 and 439, respectively, and it is recognised from Fig. 143 that 211 observations yielded negative silhouette widths.

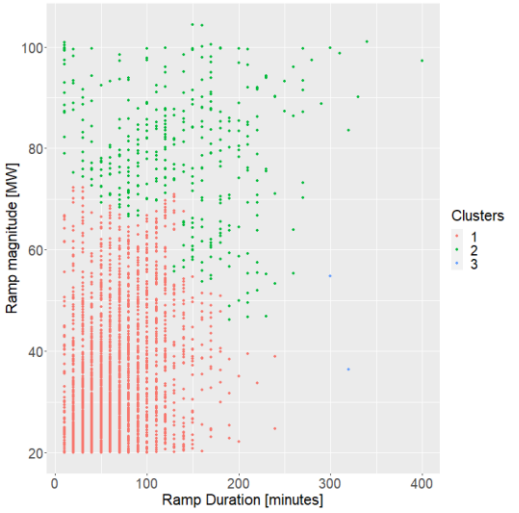


Fig. 136: Clusters obtained for the upward ramps with the agglomerative algorithm using the Euclidean distance metric and average linkage criterion.

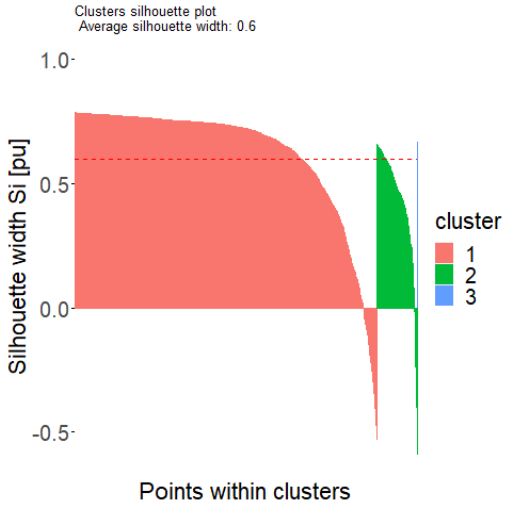


Fig. 137: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Euclidean distance metric and the average linkage criterion.

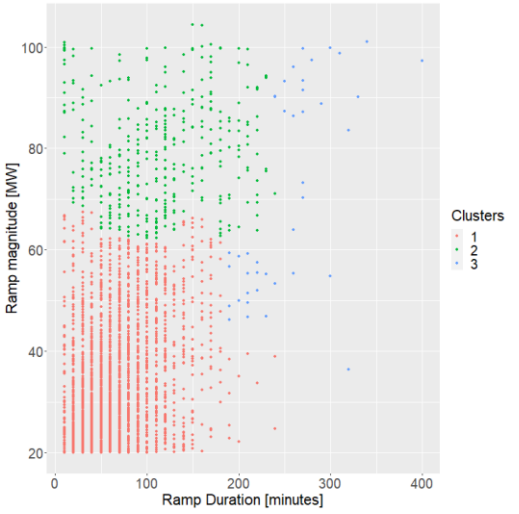


Fig. 138: Clusters obtained for the upward ramps with the agglomerative algorithm using the Euclidean distance metric and complete linkage criterion.

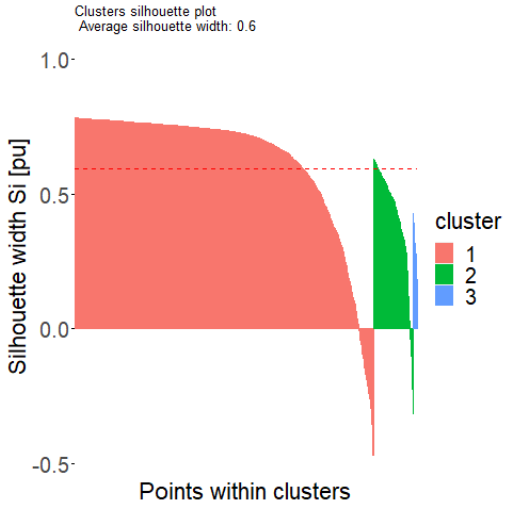


Fig. 139: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Euclidean distance metric and the complete linkage criterion.

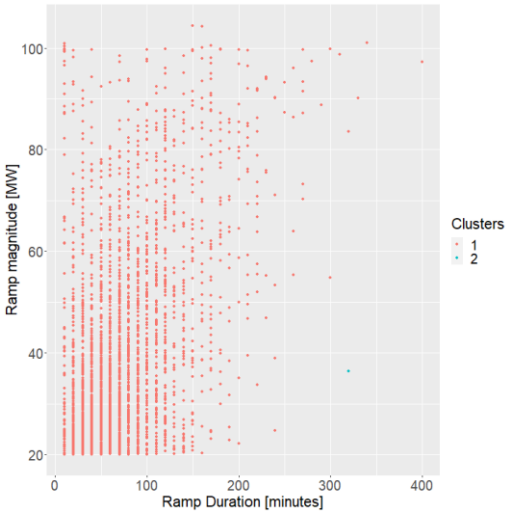


Fig. 140: Clusters obtained for the upward ramps with the agglomerative algorithm using the Euclidean distance metric and single linkage criteria.

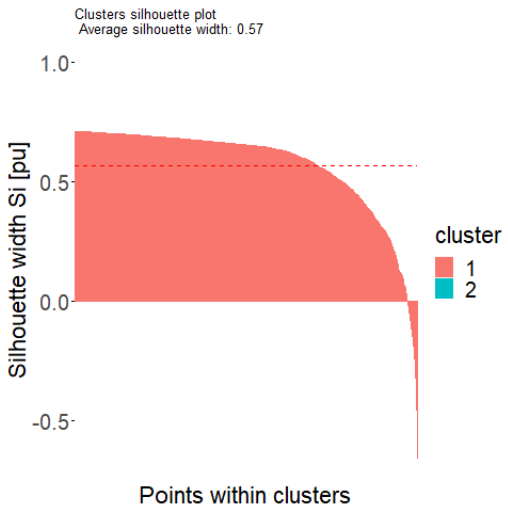


Fig. 141: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Euclidean distance metric and the single linkage criterion.

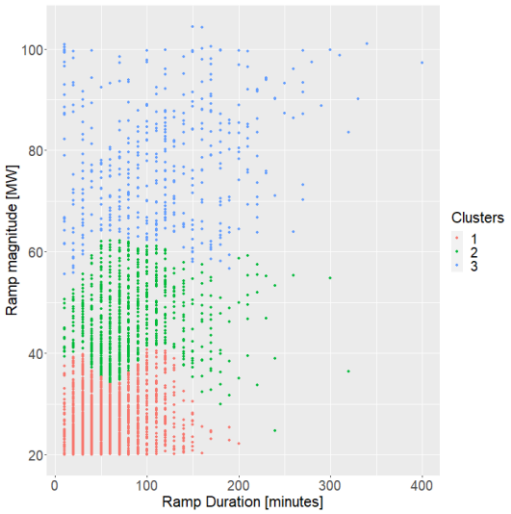


Fig. 142: Clusters obtained for the upward ramps with the agglomerative algorithm using the Euclidean distance metric and Ward linkage criterion.

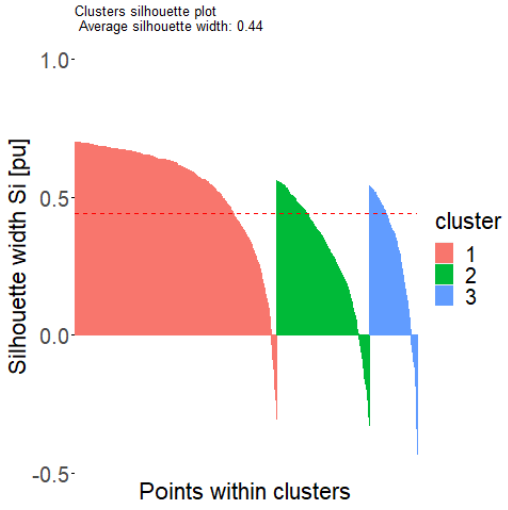


Fig. 143: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Euclidean distance metric and the Ward linkage criterion.

Fig. 288 to Fig. 291 in Appendix C show a dendrogram of the results of the agglomerative algorithm with the Manhattan distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the upward ramps.

The non-overlapping clusters obtained for the detected upward ramps with the agglomerative algorithm for the Manhattan distance metrics and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 144, Fig. 146, Fig. 148 and Fig. 150. The silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Manhattan distance metrics and average,

complete, single and Ward linkage criteria, respectively, is shown in Fig. 145, Fig. 147, Fig. 149 and Fig. 151.

Fig. 144 shows that clustering the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the average linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 2544 and 574, respectively, and it is recognised from Fig. 145 that 57 observations yielded negative silhouette widths. Fig. 146 shows that clustering the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the complete linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 3092 and 26, respectively, and it is recognised from Fig. 147 that 221 observations yielded negative silhouette widths. Fig. 148 shows that clustering the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the single linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 3117 and 1, respectively, and it is recognised from Fig. 149 that 96 observations yielded negative silhouette widths. Fig. 150 shows that clustering the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the Ward linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 2368 and 750, respectively, and it is recognised from Fig. 151 that 91 observations yielded negative silhouette widths.

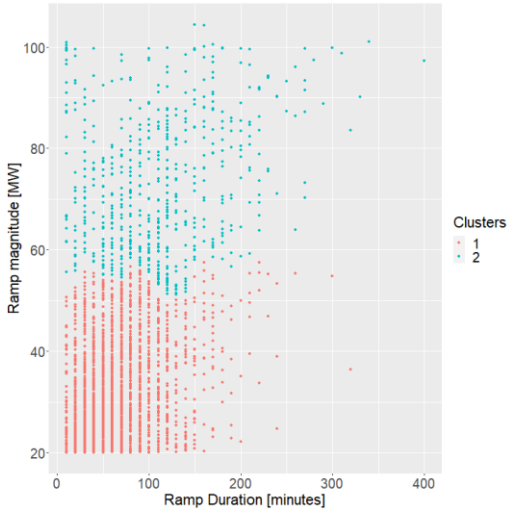


Fig. 144: Clusters obtained for the upward ramps with the agglomerative algorithm using the Manhattan distance metric and average linkage criterion.

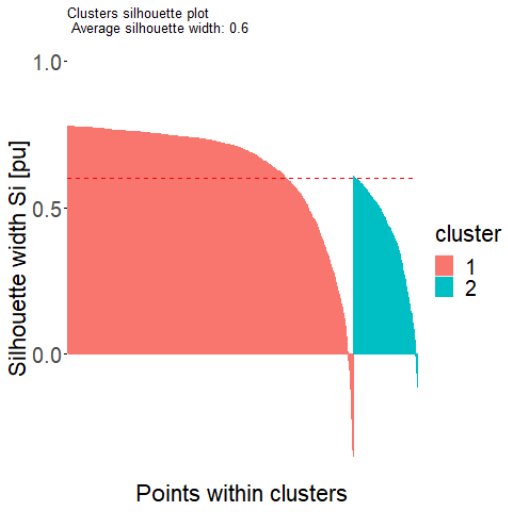


Fig. 145: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Manhattan distance metric and the average linkage criterion.

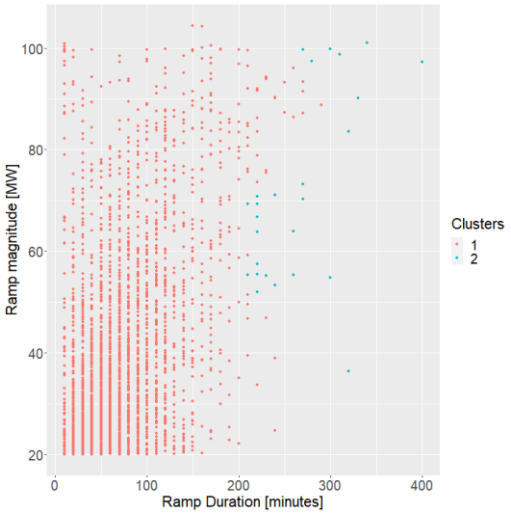


Fig. 146: Clusters obtained for the upward ramps with the agglomerative algorithm using the Manhattan distance metric and complete linkage criterion.

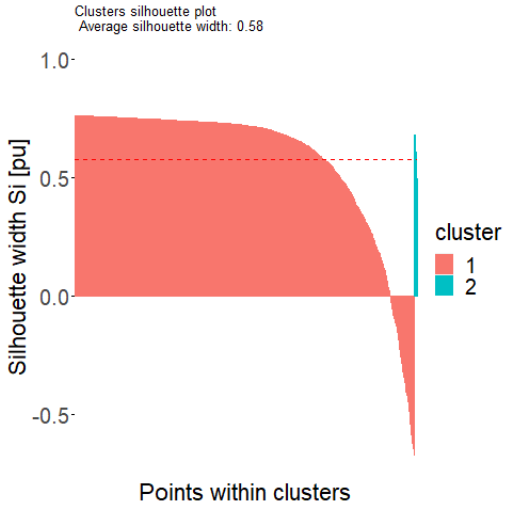


Fig. 147: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Manhattan distance metric and the complete linkage criterion.

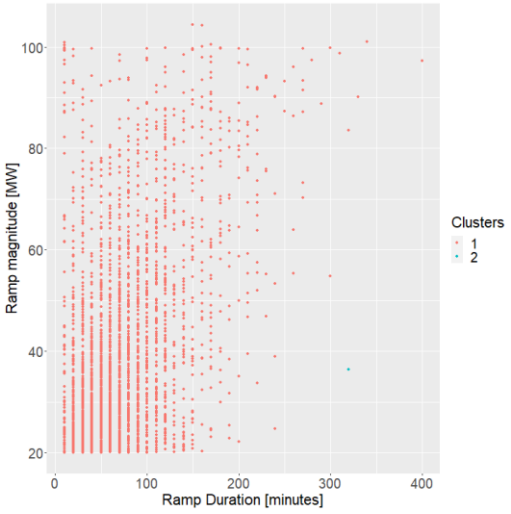


Fig. 148: Clusters obtained for the upward ramps with the agglomerative algorithm using the Manhattan distance metric and single linkage criterion.

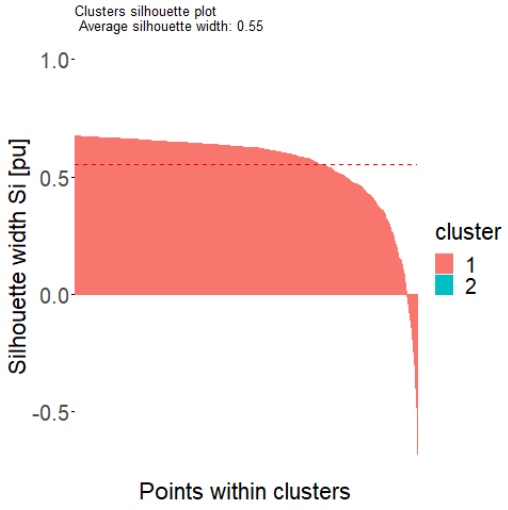


Fig. 149: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Manhattan distance metric and the single linkage criterion.

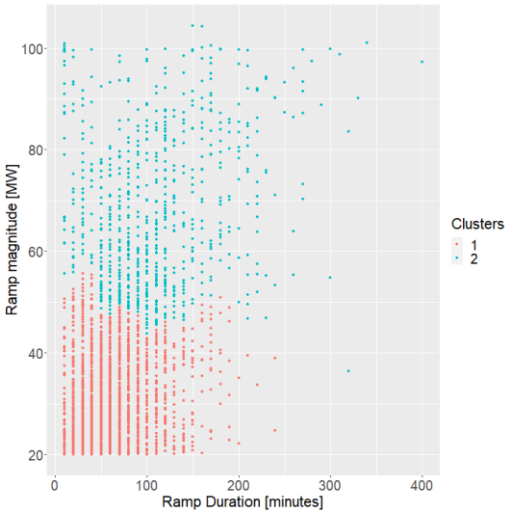


Fig. 150: Clusters obtained for the upward ramps with the agglomerative algorithm using the Manhattan distance metric and Ward linkage criterion.

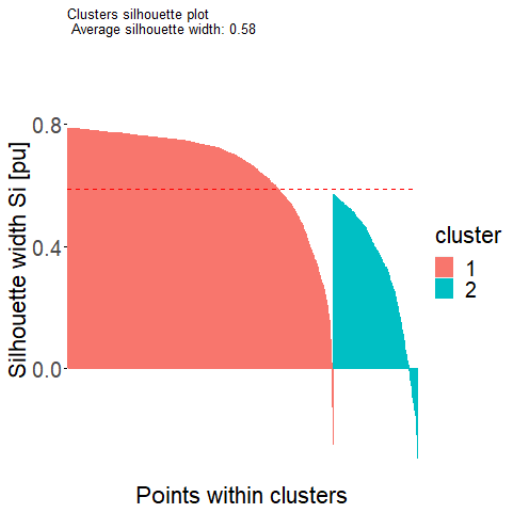


Fig. 151: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the Manhattan distance metric and the Ward linkage criterion.

Fig. 292 to Fig. 295 in Appendix C show a dendrogram of the results of the agglomerative algorithm with the maximum distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the upward ramps.

The non-overlapping clusters obtained for the detected upward ramps with the agglomerative algorithm for the maximum distance metrics and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 152, Fig. 154, Fig. 156 and Fig. 158. The silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the maximum distance metrics and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 153, Fig. 155, Fig. 157 and Fig. 159.

Fig. 152 shows that clustering the upward ramps with the agglomerative algorithm using the maximum distance metric and the average linkage criterion and cutting the resulting dendrogram to yield 3 groups resulted in 3 cluster sizes of 5267, 544 and 7, respectively, and it is recognised from Fig. 153 that 86 observations yielded negative silhouette widths. Fig. 154 shows that clustering the upward ramps with the agglomerative algorithm using the maximum distance metric and the complete linkage criterion and cutting the resulting dendrogram to yield 3 groups resulted in 3 cluster sizes of 2694, 406 and 18, respectively, and it is recognised from Fig. 155 that 212 observations yielded negative silhouette widths. Fig. 156 shows that clustering the upward ramps with the agglomerative algorithm using the maximum distance metric and the single linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 3117 and 1, respectively, and it is recognised from Fig. 157 that 84 observations yielded negative silhouette widths. Fig. 158 shows that clustering the upward ramps with the agglomerative algorithm using the maximum distance metric and the Ward linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 2577 and 541, respectively, and it is recognised from Fig. 159 that 71 observations yielded negative silhouette widths.

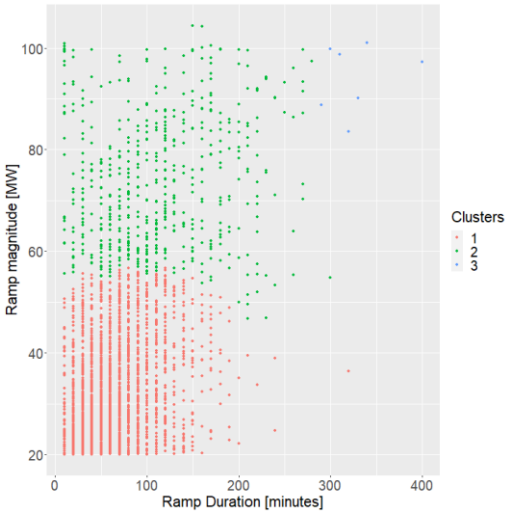


Fig. 152: Clusters obtained for the upward ramps with the agglomerative algorithm using the maximum distance metric and average linkage criterion.

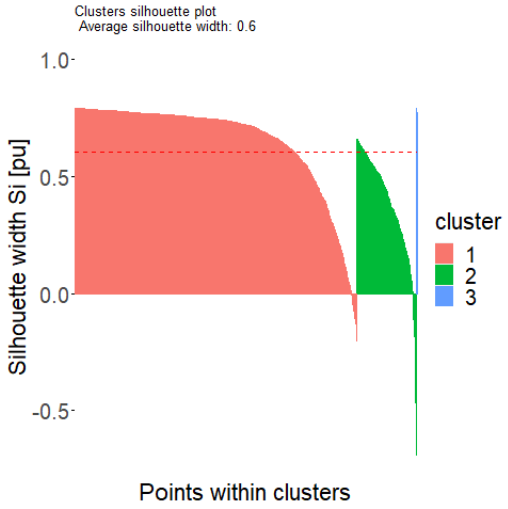


Fig. 153: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the maximum distance metric and the average linkage criterion.

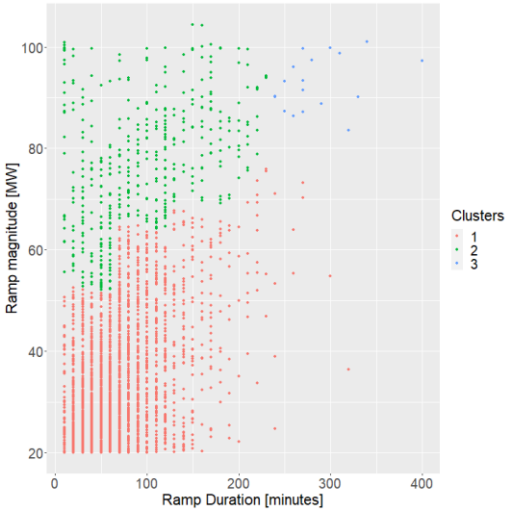


Fig. 154: Clusters obtained for the upward ramps with the agglomerative algorithm using the maximum distance metric and complete linkage criterion.

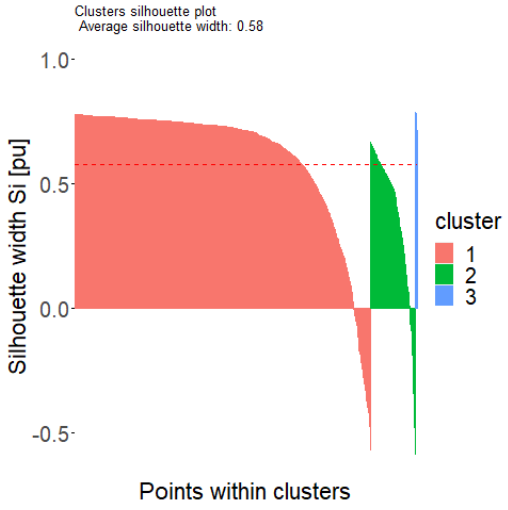


Fig. 155: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the maximum distance metric and the complete linkage criterion.

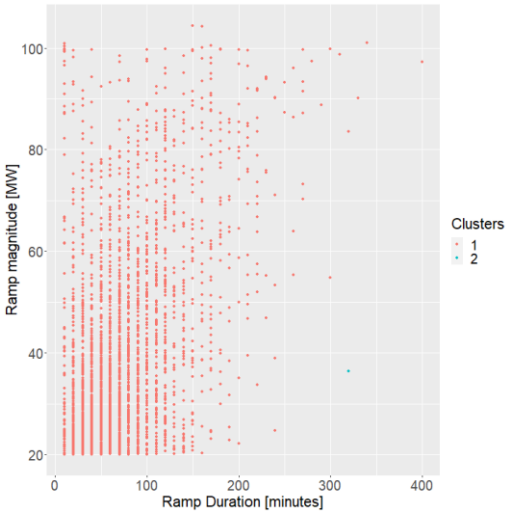


Fig. 156: Clusters obtained for the upward ramps with the agglomerative algorithm using the maximum distance metric and single linkage criterion.

Fig. 157: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the maximum distance metric and the single linkage criterion.

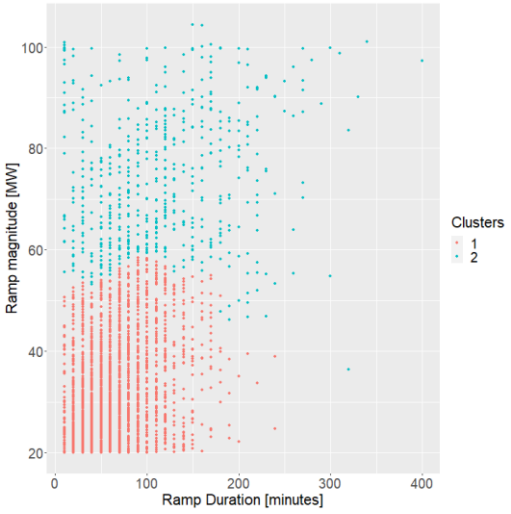


Fig. 158: Clusters obtained for the upward ramps with the agglomerative algorithm using the maximum distance metric and Ward linkage criterion.

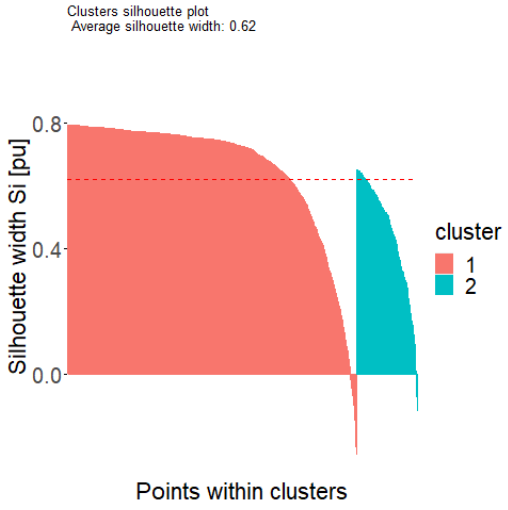


Fig. 159: Silhouette information according to the clustering of the upward ramps with the agglomerative algorithm for the maximum distance metric and the Ward linkage criterion.

5.3.4.2 Downward Ramps

Fig. 296 to Fig. 299 in Appendix C show a dendrogram of the results of the agglomerative algorithm with the Euclidean distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the downward ramps.

The non-overlapping clusters obtained for the detected downward ramps with the agglomerative algorithm for the Euclidean distance metrics and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 160, Fig. 162, Fig. 164 and Fig. 166. The silhouette information according to

the clustering of the downward ramps with the agglomerative algorithm for the Euclidean distance metrics and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 161, Fig. 163, Fig. 165 and Fig. 167.

Fig. 160 shows that clustering the downward ramps with the agglomerative algorithm using the Euclidean distance metric and the average linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 3030 and 158, respectively, and it is recognised from Fig. 161 that 190 observations yielded negative silhouette widths. Fig. 162 shows that clustering the downward ramps with the agglomerative algorithm using the Euclidean distance metric and the complete linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 2736 and 452, respectively, and it is recognised from Fig. 163 that 96 observations yielded negative silhouette widths. Fig. 164 shows that clustering the downward ramps with the agglomerative algorithm using the Euclidean distance metric and the single linkage criterion and cutting the resulting dendrogram to yield 7 groups resulted in 7 cluster sizes of 3172, 3, 8, 1, 2, 1 and 1, respectively, and it is recognised from Fig. 165 that 356 observations yielded negative silhouette widths. Fig. 166 shows that clustering the downward ramps with the agglomerative algorithm using the Euclidean distance metric and the Ward linkage criterion and cutting the resulting dendrogram to yield 3 groups resulted in 3 cluster sizes of 2165, 254 and 769, respectively, and it is recognised from Fig. 167 that 276 observations yielded negative silhouette widths.

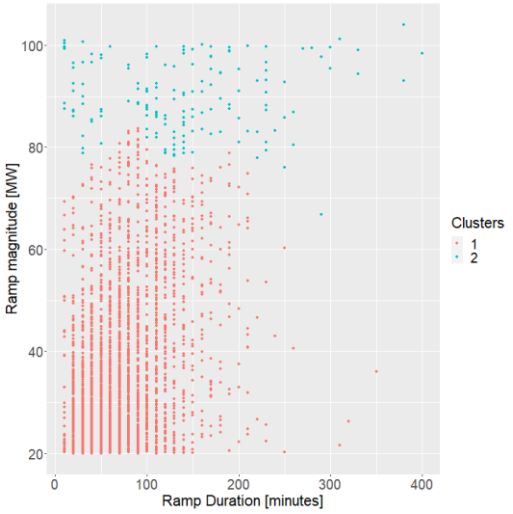


Fig. 160: Clusters obtained for the downward ramps with the agglomerative algorithm using the Euclidean distance metric and average linkage criterion.

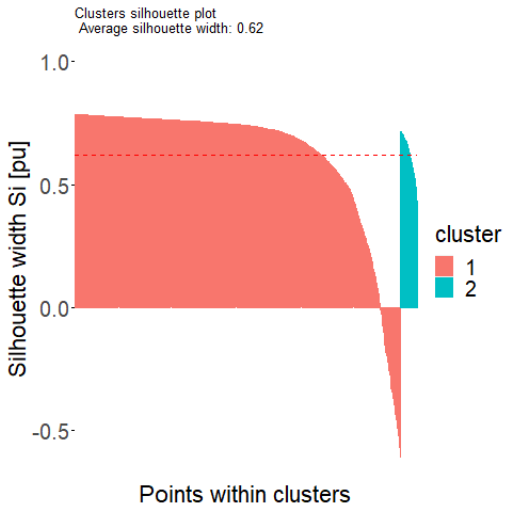


Fig. 161: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Euclidean distance metric and the average linkage criterion.

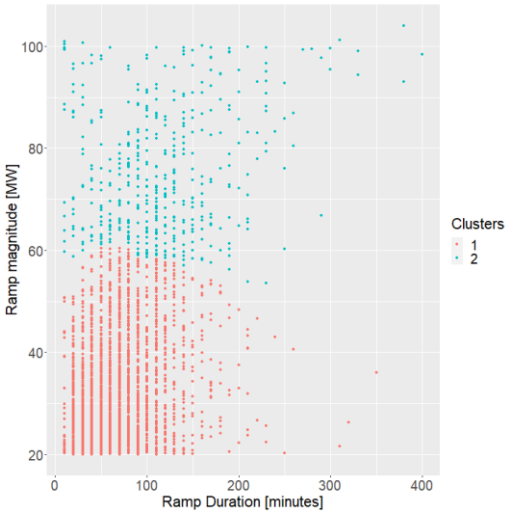


Fig. 162: Clusters obtained for the downward ramps with the agglomerative algorithm using the Euclidean distance metric and complete linkage criterion.

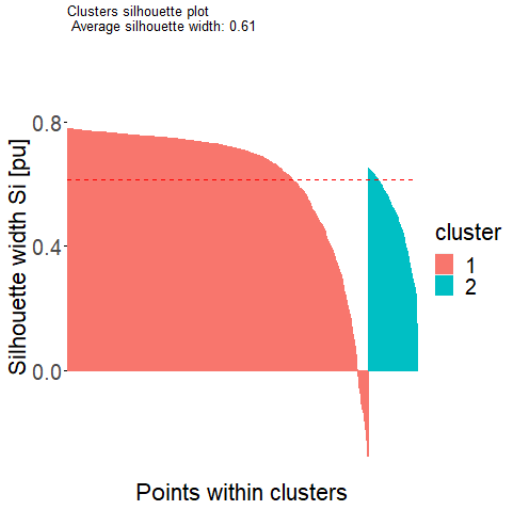


Fig. 163: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Euclidean distance metric and the complete linkage criterion.

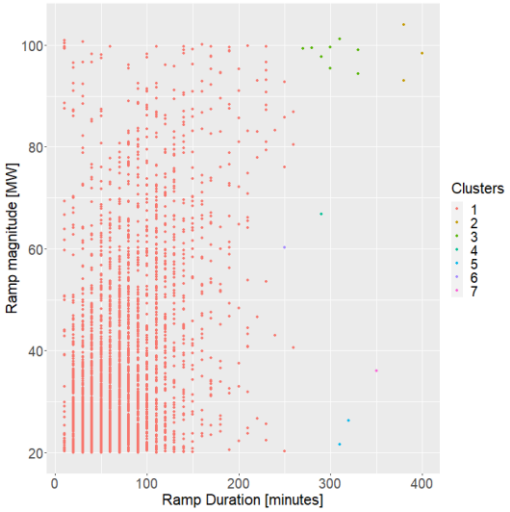


Fig. 164: Clusters obtained for the downward ramps with the agglomerative algorithm using the Euclidean distance metric and single linkage criterion.

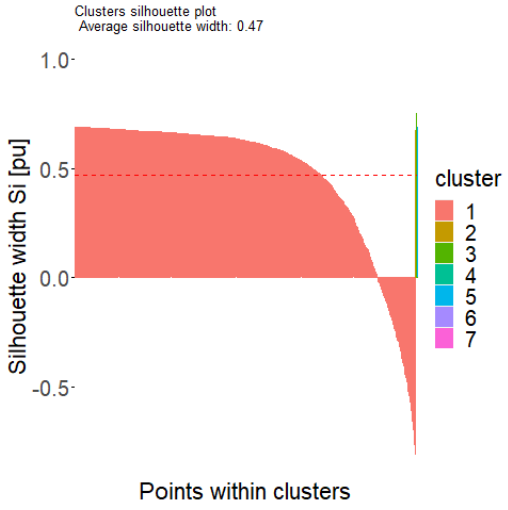


Fig. 165: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Euclidean distance metric and the single linkage criterion.

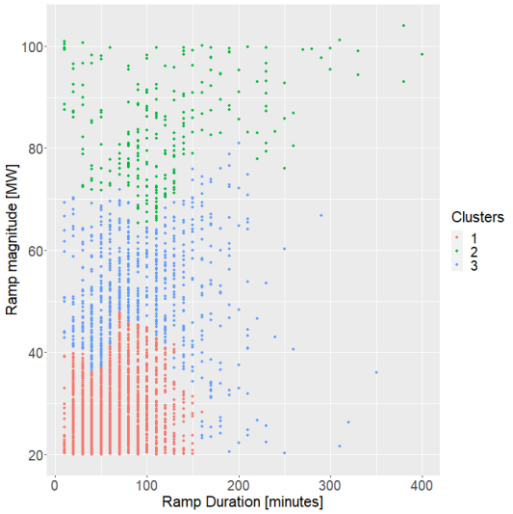


Fig. 166: Clusters obtained for the downward ramps with the agglomerative algorithm using the Euclidean distance metric and Ward linkage criterion.

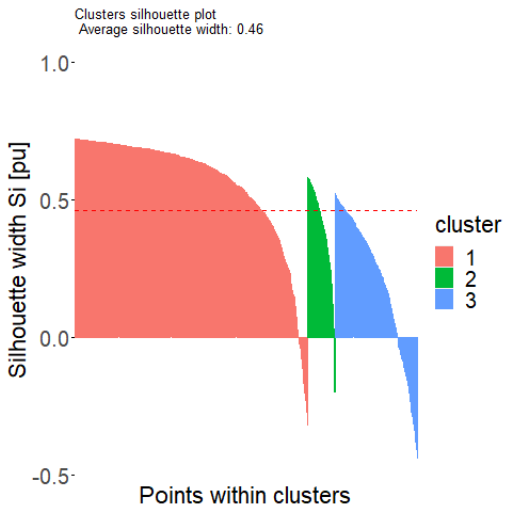


Fig. 167: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Euclidean distance metric and the Ward linkage criterion.

Fig. 300 to Fig. 303 in Appendix C show a dendrogram of the results of the agglomerative algorithm with the Manhattan distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the downward ramps.

The non-overlapping clusters obtained for the detected downward ramps with the agglomerative algorithm for the Manhattan distance metric and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 168, Fig. 170, Fig. 172 and Fig. 174. The silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Manhattan distance metrics and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 169, Fig. 171, Fig. 173 and Fig. 175.

Fig. 168 shows that clustering the downward ramps with the agglomerative algorithm using the Manhattan distance metric and the average linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 2928 and 260, respectively, and it is recognised from Fig. 169 that 150 observations yielded negative silhouette widths. Fig. 170 shows that clustering the downward ramps with the agglomerative algorithm using the Manhattan distance metric and the complete linkage criterion and cutting the resulting dendrogram to yield 6 groups resulted in 6 cluster sizes of 1705, 961, 290, 153, 38 and 41, respectively, and it is recognised from Fig. 171 that 368 observations yielded negative silhouette widths. Fig. 172 shows that clustering the downward ramps with the agglomerative algorithm using the Manhattan distance metric and the single linkage criterion and cutting the resulting dendrogram to yield 9 groups resulted in 9 cluster sizes of 3172, 1, 8, 1, 1, 2, 1, 1 and 1, respectively, and it is recognised from Fig. 173 that 361 observations yielded negative silhouette widths. Fig. 174 shows that clustering the downward ramps with the agglomerative algorithm using the Manhattan distance metric and the Ward linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 2546 and 642, respectively, and it is recognised from Fig. 175 that 53 observations yielded negative silhouette widths.

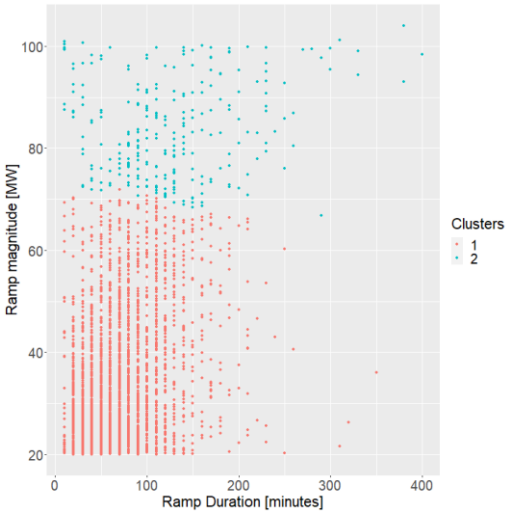


Fig. 168: Clusters obtained for the downward ramps with the agglomerative algorithm using the Manhattan distance metric and average linkage criterion.

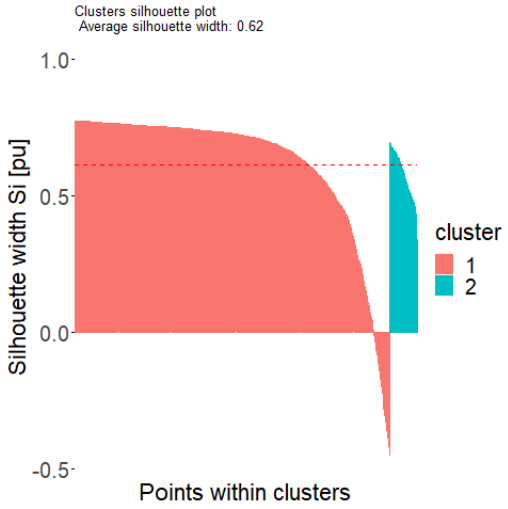


Fig. 169: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Manhattan distance metric and the average linkage criterion.

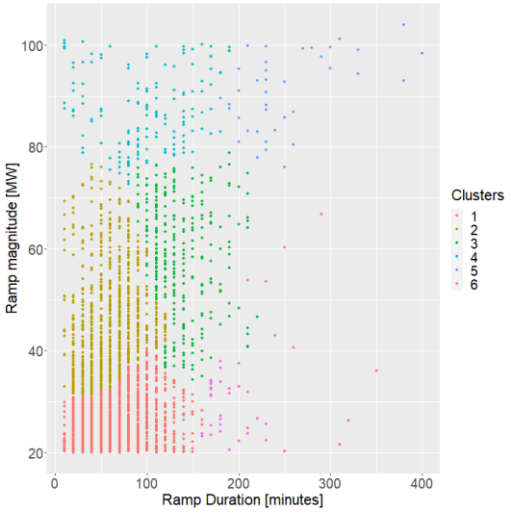


Fig. 170: Clusters obtained for the downward ramps with the agglomerative algorithm using the Manhattan distance metric and complete linkage criterion.

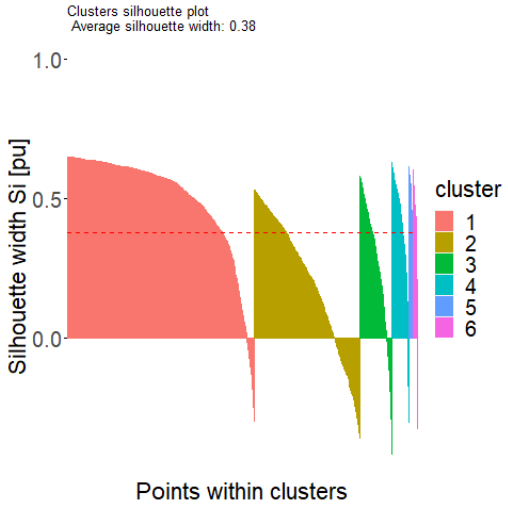


Fig. 171: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Manhattan distance metric and the complete linkage criterion.

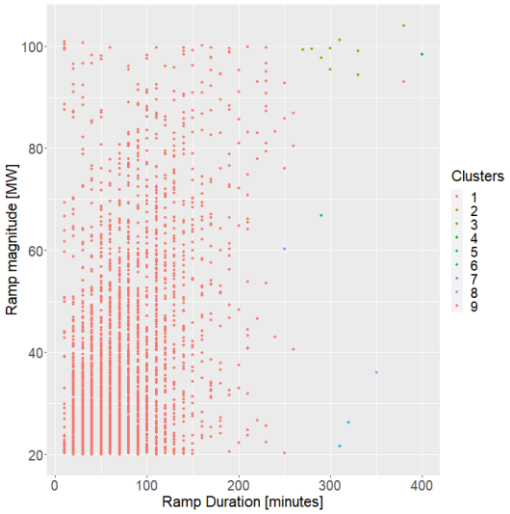


Fig. 172: Clusters obtained for the downward ramps with the agglomerative algorithm using the Manhattan distance metric and single linkage criterion.

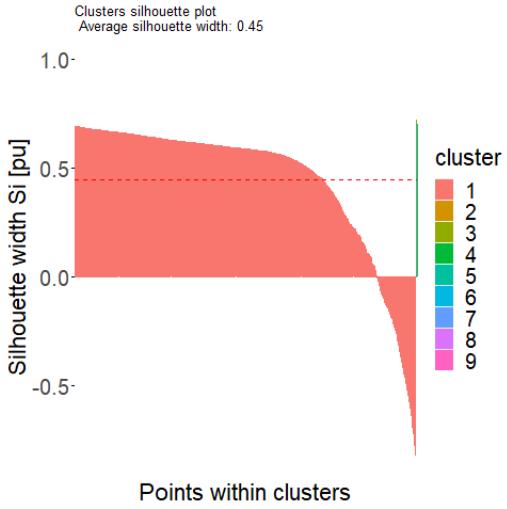


Fig. 173: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Manhattan distance metric and the single linkage criterion.

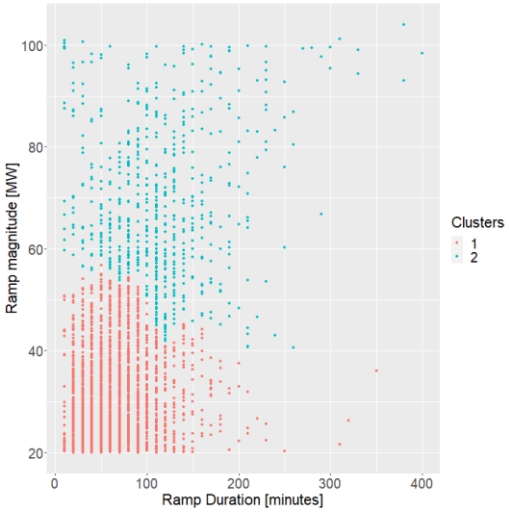


Fig. 174: Clusters obtained for the downward ramps with the agglomerative algorithm using the Manhattan distance metric and Ward linkage criterion.

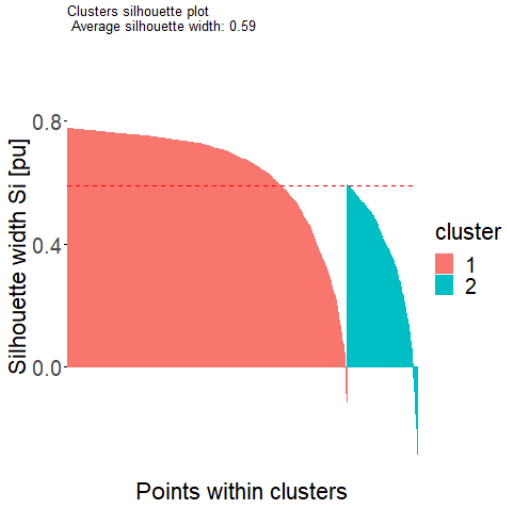


Fig. 175: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the Manhattan distance metric and the Ward linkage criterion.

Fig. 304 to Fig. 307 in Appendix C show a dendrogram of the results of the agglomerative algorithm with the maximum distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the downward ramps.

The non-overlapping clusters obtained for the detected downward ramps with the agglomerative algorithm for the maximum distance metrics and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 176, Fig. 178, Fig. 180 and Fig. 182. The silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the maximum distance

metrics and average, complete, single and Ward linkage criteria, respectively, is shown in Fig. 177, Fig. 179, Fig. 181 and Fig. 183.

Fig. 176 shows that clustering the downward ramps with the agglomerative algorithm using the maximum distance metric and the average linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 2881 and 307, respectively, and it is recognised from Fig. 177 that 161 observations yielded negative silhouette widths. Fig. 178 shows that clustering the downward ramps with the agglomerative algorithm using the maximum distance metric and the complete linkage criterion and cutting the resulting dendrogram to yield 5 groups resulted in 5 cluster sizes of 2175, 211, 609, 46 and 147, respectively, and it is recognised from Fig. 179 that 360 observations yielded negative silhouette widths. Fig. 180 shows that clustering the downward ramps with the agglomerative algorithm using the maximum distance metric and the single linkage criterion and cutting the resulting dendrogram to yield 7 groups resulted in 7 cluster sizes of 3172, 3, 8, 1, 2, 1 and 1, respectively, and it is recognised from Fig. 181 that 360 observations yielded negative silhouette widths. Fig. 182 shows that clustering the downward ramps with the agglomerative algorithm using the maximum distance metric and the Ward linkage criterion and cutting the resulting dendrogram to yield 2 groups resulted in 2 cluster sizes of 1454, 349 and 1385, respectively, and it is recognised from Fig. 183 that 468 observations yielded negative silhouette widths.

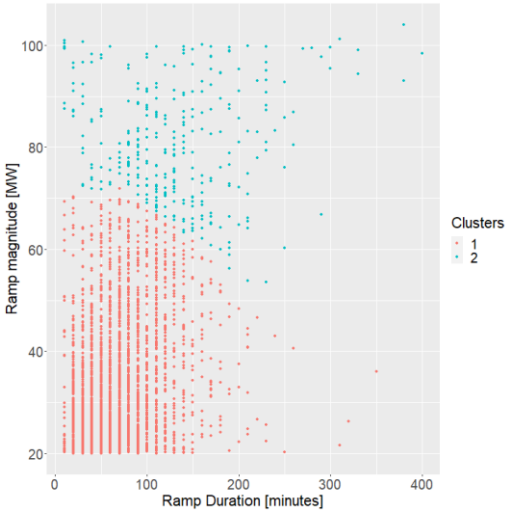


Fig. 176: Clusters obtained for the downward ramps with the agglomerative algorithm using the maximum distance metric and average linkage criterion.

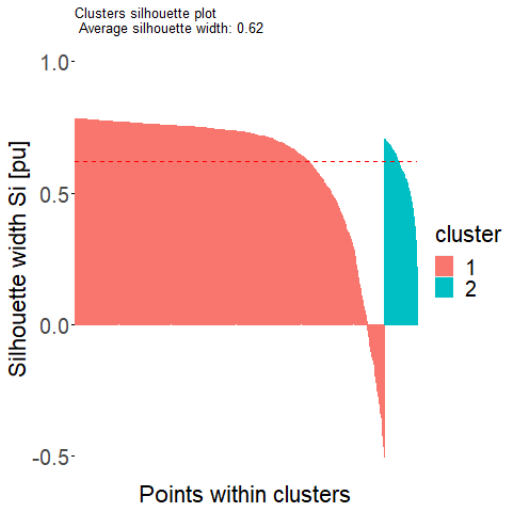


Fig. 177: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the maximum distance metric and the average linkage criterion.

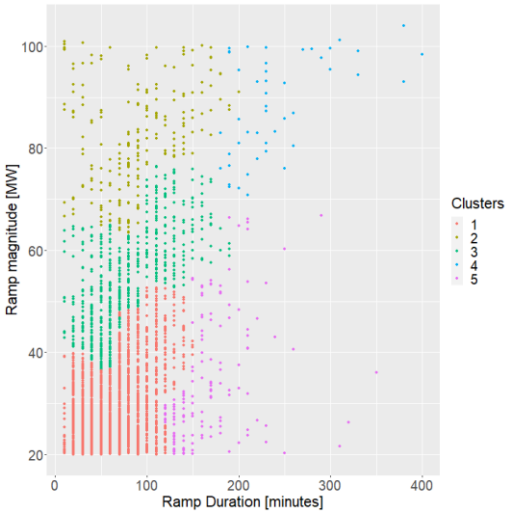


Fig. 178: Clusters obtained for the downward ramps with the agglomerative algorithm using the maximum distance metric and complete linkage criterion.

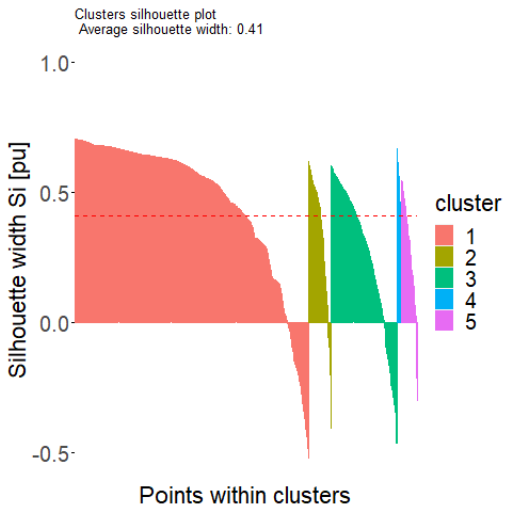


Fig. 179: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the maximum distance metric and the complete linkage criterion.

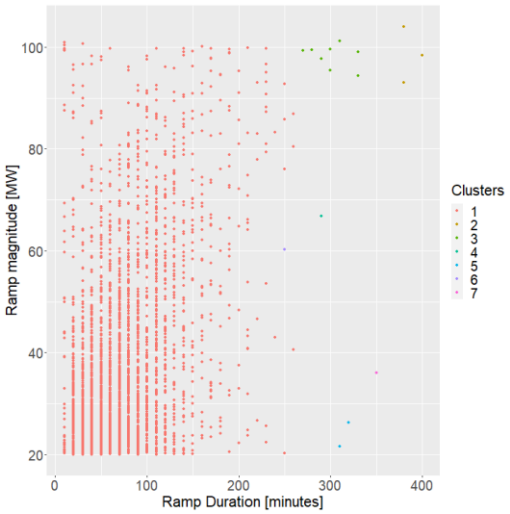


Fig. 180: Clusters obtained for the downward ramps with the agglomerative algorithm using the maximum distance metric and single linkage criterion.

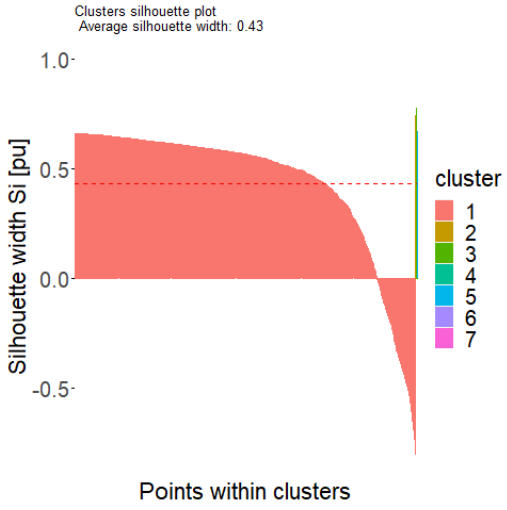


Fig. 181: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the maximum distance metric and the single linkage criterion.

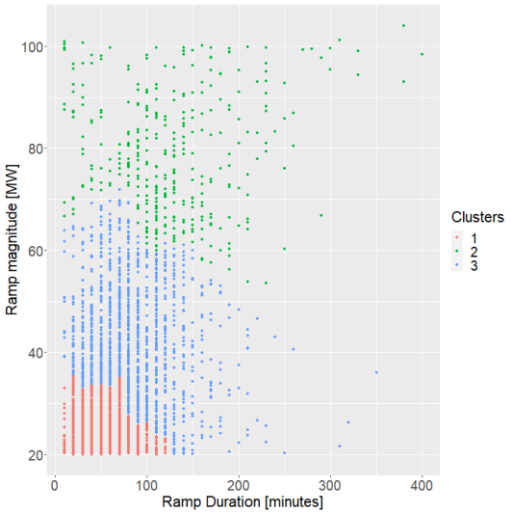


Fig. 182: Clusters obtained for the downward ramps with the agglomerative algorithm using the maximum distance metric and Ward linkage criterion.

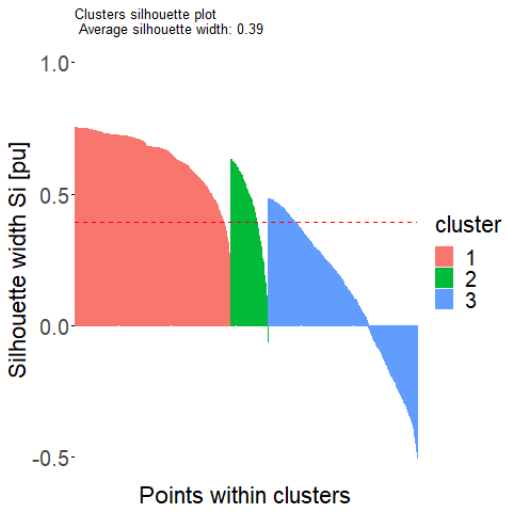


Fig. 183: Silhouette information according to the clustering of the downward ramps with the agglomerative algorithm for the maximum distance metric and the Ward linkage criterion.

5.3.5 Results for divisive analysis clustering

5.3.5.1 Upward ramps

The non-overlapping clusters obtained for the detected upward ramps with the DIANA algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 184, Fig. 186 and Fig. 188, respectively. The silhouette information according to the clustering of the upward ramps with the DIANA algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 185, Fig. 187 and Fig. 189, respectively.

Fig. 184 shows that clustering the upward ramps with the DIANA algorithm using the Euclidean distance metric into 2 groups resulted in 2 cluster sizes of 2498 and 620, respectively, and it is recognised from Fig. 185 that 0 observations yielded negative silhouette widths. Fig. 186 shows that clustering the upward ramps with the DIANA algorithm using the Manhattan distance metric into 3 groups resulted in 3 cluster sizes of 2507, 201 and 410, respectively, and it is recognised from Fig. 187 that 157 observations yielded negative silhouette widths. Fig. 188 shows that clustering the upward ramps with the DIANA algorithm using the maximum distance metric into 2 groups resulted in 2 cluster sizes of 2476 and 642, respectively, and it is recognised from Fig. 189 that 0 observations yielded negative silhouette widths.

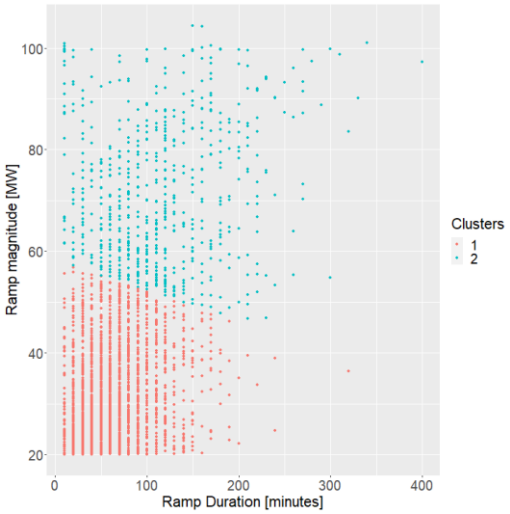


Fig. 184: Clusters obtained for the upward ramps with the DIANA algorithm for the Euclidean distance metric.

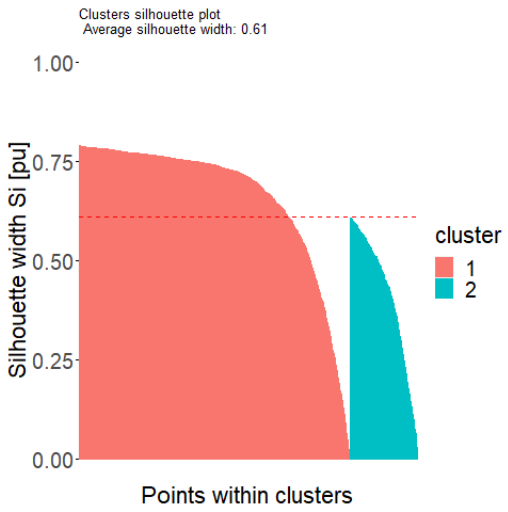


Fig. 185: Silhouette information according to the clustering of the upward ramps with the DIANA algorithm for the Euclidean distance metric.

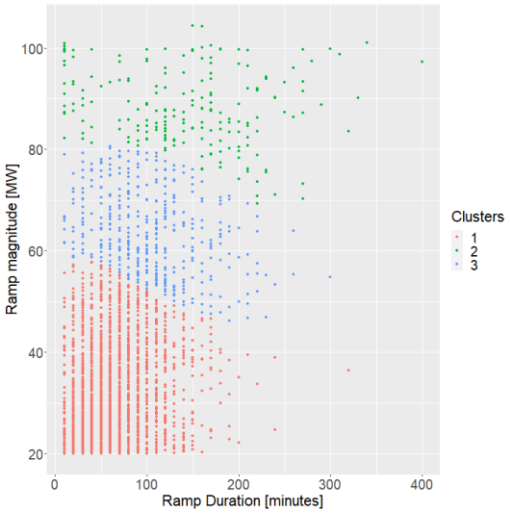


Fig. 186: Clusters obtained for the upward ramps with the DIANA algorithm for the Manhattan distance metric.

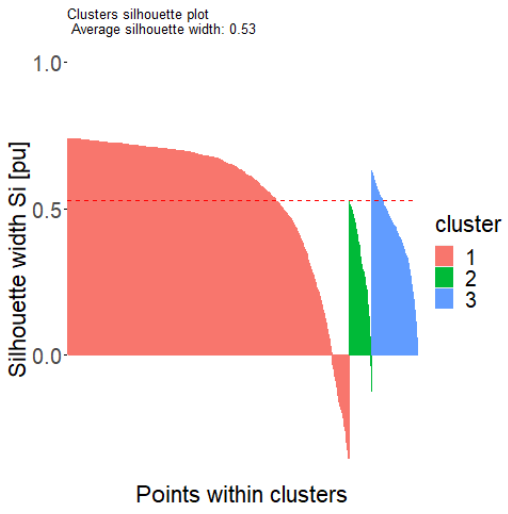


Fig. 187: Silhouette information according to the clustering of the upward ramps with the DIANA algorithm for the Manhattan distance metric.

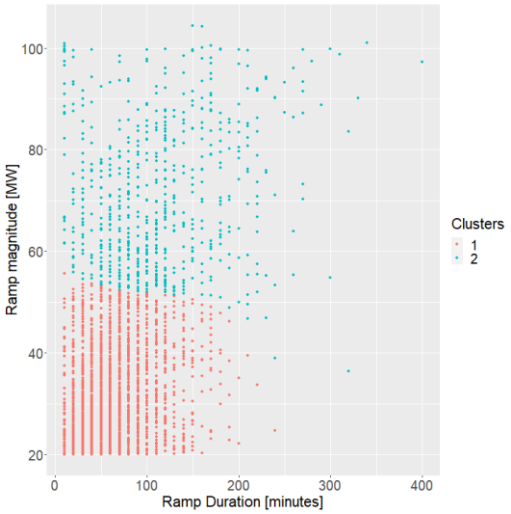


Fig. 188: Clusters obtained for the upward ramps with the DIANA algorithm for the maximum distance metric.

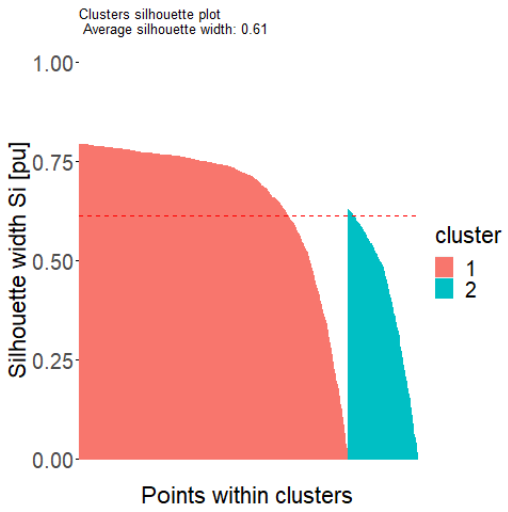


Fig. 189: Silhouette information according to the clustering of the upward ramps with the DIANA algorithm for the maximum distance metric.

5.3.5.2 Downward ramps

The non-overlapping clusters obtained for the detected downward ramps with the DIANA algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 190, Fig. 192 and Fig. 194, respectively. The silhouette information according to the clustering of the downward ramps with the DIANA algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 191, Fig. 193 and Fig. 195, respectively.

Fig. 190 shows that clustering the downward ramps with the DIANA algorithm using the Euclidean distance metric into 2 groups resulted in 2 cluster sizes of 5229 and 629, respectively, and it is recognised from Fig. 191 that 0 observations yielded negative silhouette widths. Fig. 192 shows that clustering the downward ramps with the DIANA algorithm using the Manhattan distance metric into 2 groups resulted in 2 cluster sizes of 2574 and 614, respectively, and it is recognised from Fig. 193 that 0 observations yielded negative silhouette widths. Fig. 194 shows that clustering the downward ramps with the DIANA algorithm using the maximum distance metric into 2 groups resulted in 2 cluster sizes of 2540 and 648, respectively, and it is recognised from Fig. 195 that 0 observations yielded negative silhouette widths.

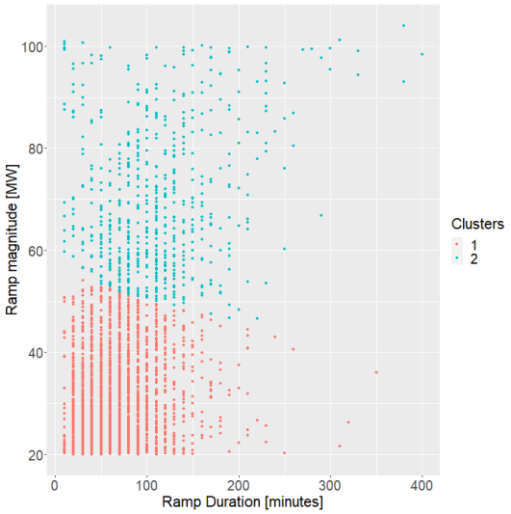


Fig. 190: Clusters obtained for the downward ramps with the DIANA algorithm for the Euclidean distance metric.

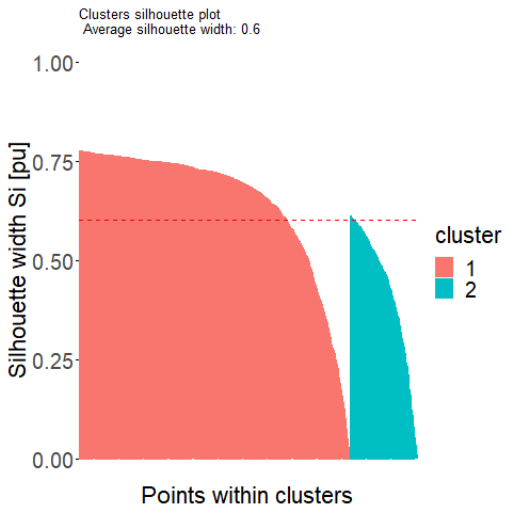


Fig. 191: Silhouette information according to the clustering of the downward ramps with the DIANA algorithm for the Euclidean distance metric.

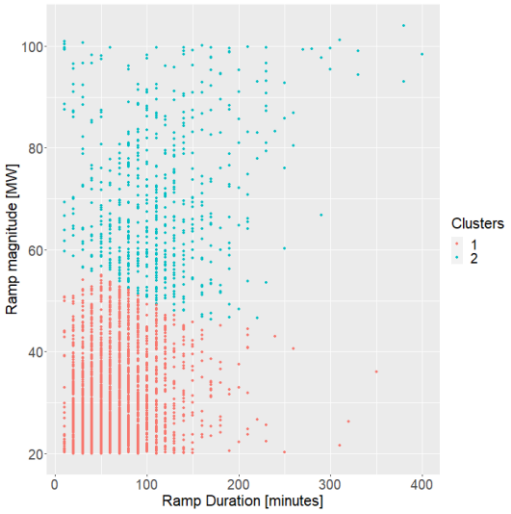


Fig. 192: Clusters obtained for the downward ramps with the DIANA algorithm for the Manhattan distance metric.

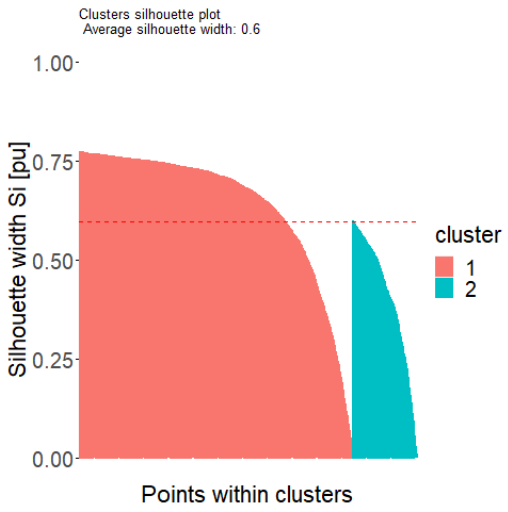


Fig. 193: Silhouette information according to the clustering of the downward ramps with the DIANA algorithm for the Manhattan distance metric.

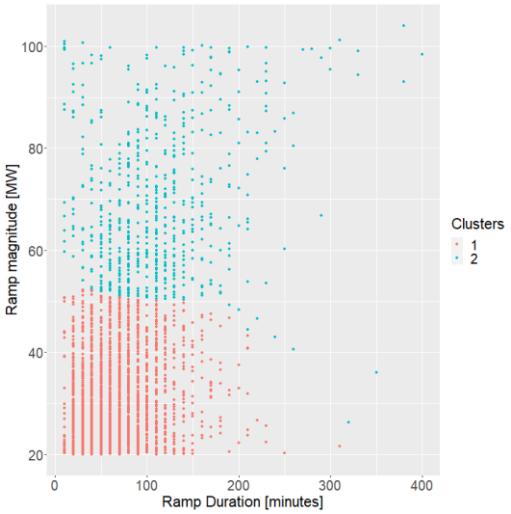


Fig. 194: Clusters obtained for the downward ramps with the DIANA algorithm for the maximum distance metric.

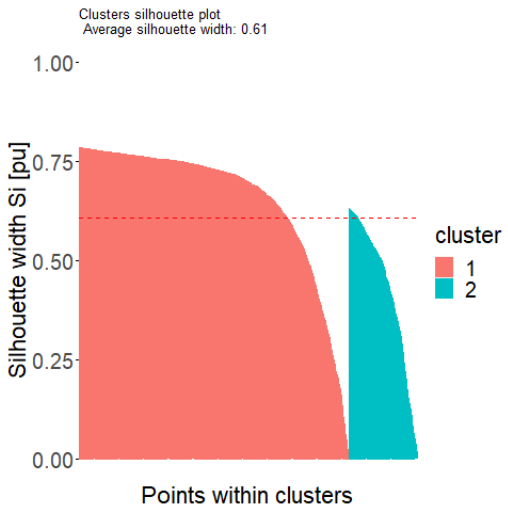


Fig. 195: Silhouette information according to the clustering of the downward ramps with the DIANA algorithm for the maximum distance metric.

5.3.6 Results for c-means clustering

5.3.6.1 Upward ramps

The non-overlapping clusters obtained for the detected upward ramps with the c-means algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 196, Fig. 198 and Fig. 200, respectively. The silhouette information according to the clustering of the upward ramps with the c-means algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 197, Fig. 199 and Fig. 201, respectively.

Fig. 196 shows that clustering the upward ramps with the c-means algorithm using the Euclidean distance metric into 2 groups resulted in 2 cluster sizes of 2040 and 1078, respectively, and it is recognised from Fig. 197 that 275 observations yielded negative silhouette widths. Fig. 198 shows that clustering the upward ramps with the c-means algorithm using the Manhattan distance metric into 2 groups resulted in 2 cluster sizes of 2033 and 1085, respectively, and it is recognised from Fig. 199 that 286 observations yielded negative silhouette widths. Fig. 200 shows that clustering the upward ramps with the c-means algorithm using the maximum distance metric into 2 groups resulted in 2 cluster sizes of 2057 and 1061, respectively, and it is recognised from Fig. 201 that 261 observations yielded negative silhouette widths.

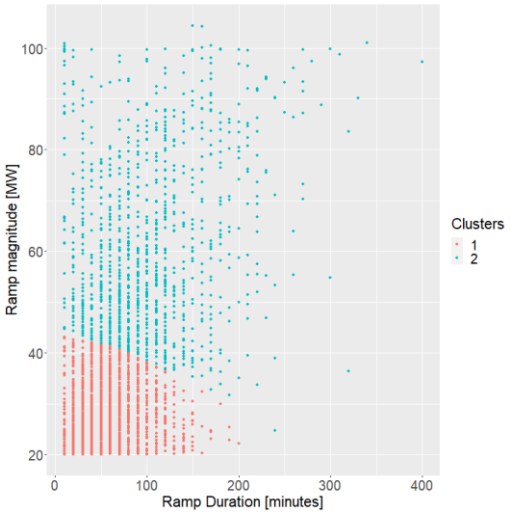


Fig. 196: Clusters obtained for the upward ramps with the c-means algorithm for the Euclidean distance metric.

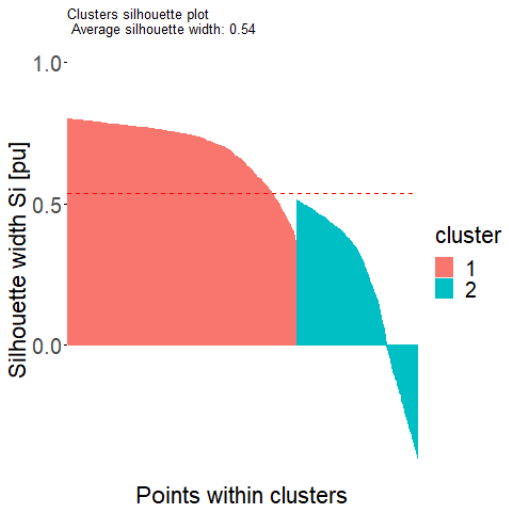


Fig. 197: Silhouette information according to the clustering of the upward ramps with the c-means algorithm for the Euclidean distance metric.

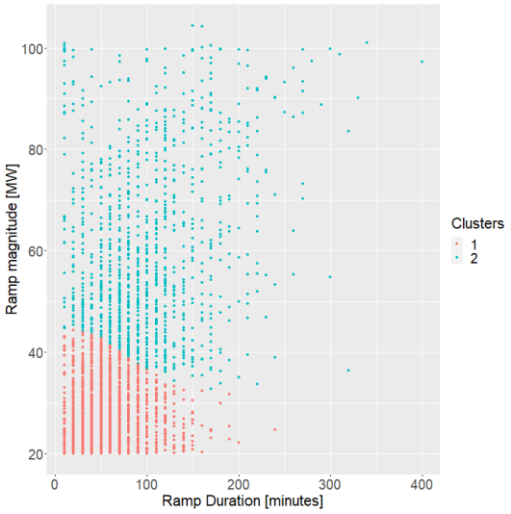


Fig. 198: Clusters obtained for the upward ramps with the c-means algorithm for the Manhattan distance metric.

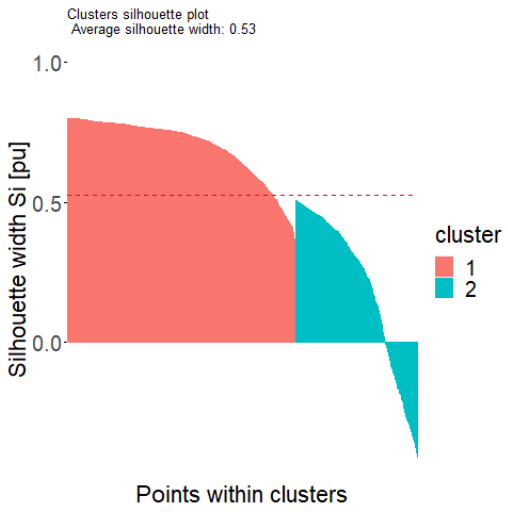


Fig. 199: Silhouette information according to the clustering of the upward ramps with the c-means algorithm for the Manhattan distance metric.

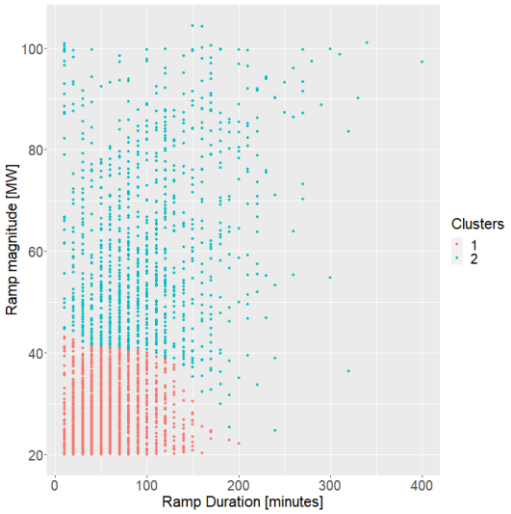


Fig. 200: Clusters obtained for the upward ramps with the c-means algorithm for the maximum distance metric.

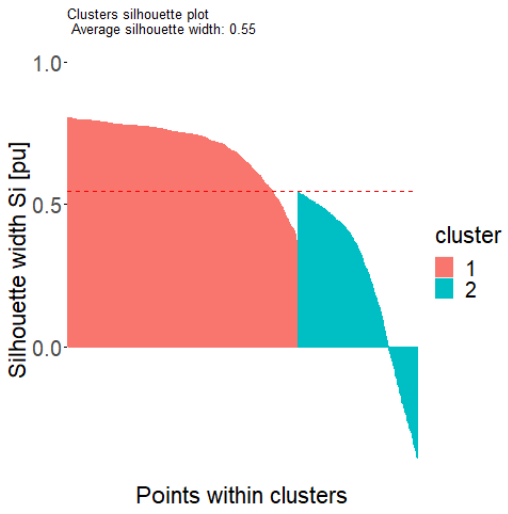


Fig. 201: Silhouette information according to the clustering of the upward ramps with the c-means algorithm for the maximum distance metric.

5.3.6.2 Downward ramps

The non-overlapping clusters obtained for the detected downward ramps with the c-means algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 202, Fig. 204 and Fig. 206, respectively. The silhouette information according to the clustering of the downward ramps with the c-means algorithm for the Euclidean, Manhattan and maximum distance metrics is shown in Fig. 203, Fig. 205 and Fig. 207, respectively.

Fig. 202 shows that clustering the downward ramps with the c-means algorithm using the Euclidean distance metric into 2 groups resulted in 2 cluster sizes of 2061 and 1127, respectively, and it is recognised from Fig. 203 that 288 observations yielded negative silhouette widths. Fig. 204 shows that clustering the downward ramps with the c-means algorithm using the Manhattan distance metric into 2 groups resulted in 2 cluster sizes of 2039 and 1149, respectively, and it is recognised from Fig. 205 that 300 observations yielded negative silhouette widths. Fig. 206 shows that clustering the downward ramps with the c-means algorithm using the maximum distance metric into 2 groups resulted in 2 cluster sizes of 2082 and 1106, respectively, and it is recognised from Fig. 207 that 194 observations yielded negative silhouette widths.

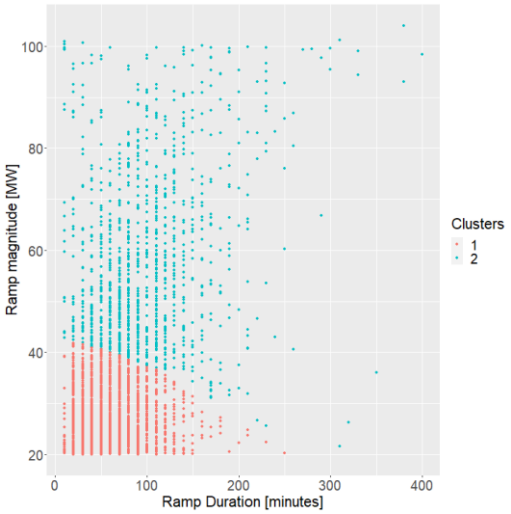


Fig. 202: Clusters obtained for the downward ramps with the c-means algorithm for the Euclidean distance metric.

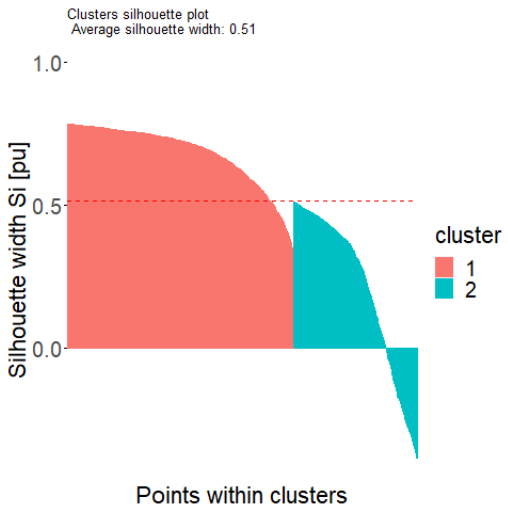


Fig. 203: Silhouette information according to the clustering of the downward ramps with the c-means algorithm for the Euclidean distance metric.

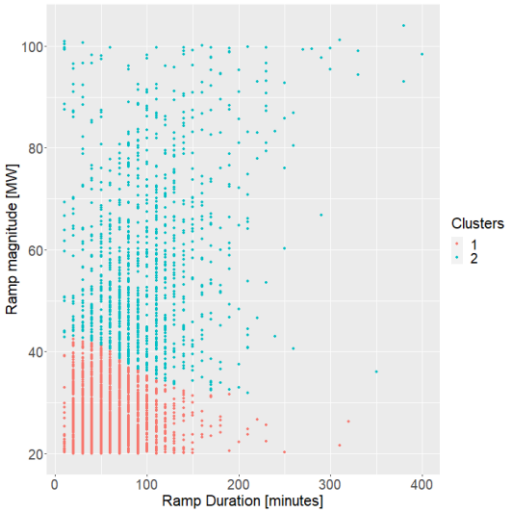


Fig. 204: Clusters obtained for the downward ramps with the c-means algorithm for the Manhattan distance metric.

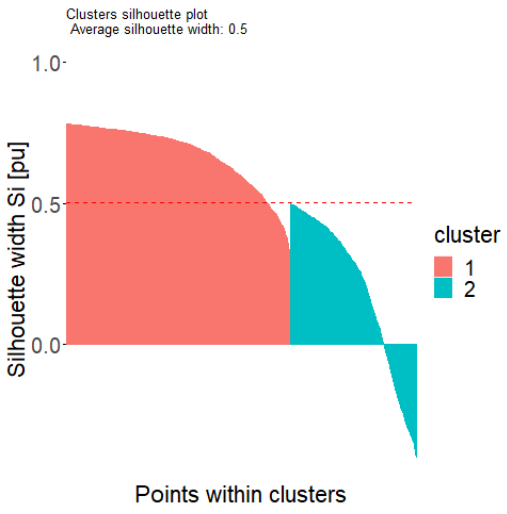


Fig. 205: Silhouette information according to the clustering of the downward ramps with the c-means algorithm for the Manhattan distance metric.

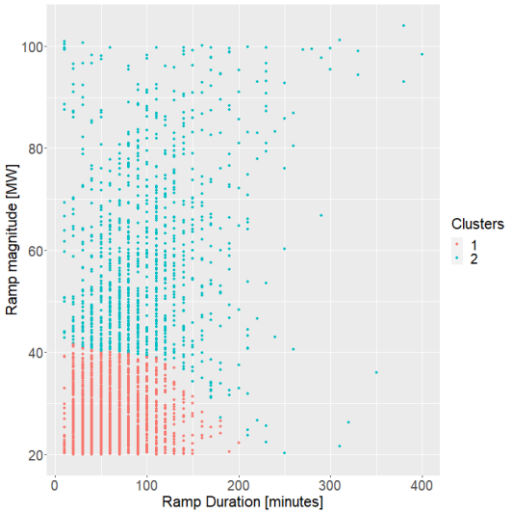


Fig. 206: Clusters obtained for the downward ramps with the c-means algorithm for the maximum distance metric.

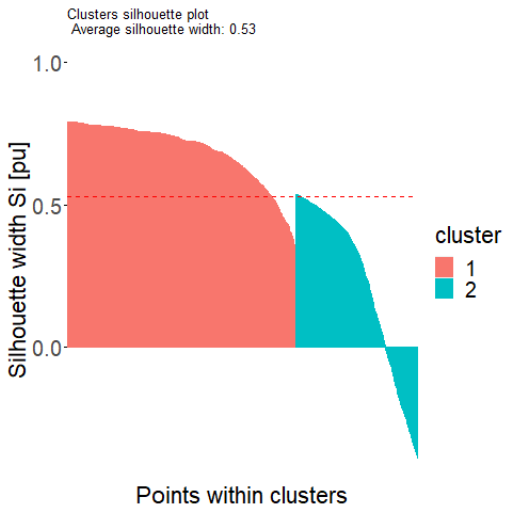


Fig. 207: Silhouette information according to the clustering of the downward ramps with the c-means algorithm for the maximum distance metric.

5.4 Comparison of validation metrics

The clustering results obtained for the detected upward and downward ramps by varying all combinations of the clustering methods, the distance measures and the linkage criteria, where applicable, differs significantly. Therefore, it is vital to evaluate the respective clustering results in order to determine whether the obtained cluster configurations are acceptable or not, as well as to compare the different cluster results so to determine the optimal clustering procedure for the dataset of interest. For this purpose, several validation measures are considered in this study, namely:

- *Silhouette coefficient:* The silhouette coefficient, i.e. average silhouette value of all the observations in a cluster, indicates how dense the observations in the cluster is. The silhouette coefficient should be as high as possible for optimal results.
- *Number of incorrect cluster assignments:* The number of incorrect cluster assignment should be minimised for optimal results. The silhouette value indicates whether an observation is incorrectly assigned to a cluster and if it is necessary to modify the cluster configuration or technique.
- *Calinski- Harabasz index:* This index makes use of the average between clusters and the within cluster sum of squares to evaluate the validity of the clusters. The Caliński-Harabasz index should be minimised for optimal clustering results.
- *Average distance within clusters:* The average distance within clusters should be minimised since this indicates close similarity between the observations in the cluster.
- *Dunn index:* The Dunn index is proportional to the inter-cluster distances, i.e. the distances between the distinct clusters, and inversely proportional to intra-cluster diameters. The Dunn index should be maximised to obtain optimal clustering results, since well separated clusters are characterised by large, large inter-cluster distance and small intra-cluster diameters.

Table 21 and Table 22 list the results of the validation measures computed for the various clustering methodologies applied to the datasets for the upward and downward ramps, respectively. The best and second-best validation results for each validation measure are indicated with blue and green, respectively. The DIANA clustering algorithm using the maximum distance measure provided the optimal clustering

results for the upward ramps, since it simultaneously yielded the minimum number of incorrect cluster assignments and the maximum silhouette coefficient. The agglomerative clustering algorithm using the Euclidean distance measure and single linkage criterion also provided good clustering results for the upward ramps, since it simultaneously yielded the second highest Calinski-Harabasz index and second highest Dunn index. The agglomerative clustering algorithm using the Euclidean distance measure and average linkage criterion provided the optimal clustering results for the downward ramps, since it simultaneously yielded the highest silhouette coefficient and highest Dunn index. Therefore, it is concluded that the hierarchical clustering algorithms outperforms the partitioning and fuzzy c-means clustering methods for the datasets of interest.

Table 21: Comparison of the validation metrics for the various clustering methods as applied to upward ramps.

Clustering algorithm	Distance metric	Linkage criterion	Optimal number of clusters	Silhouette coefficient	Number of incorrect cluster assignments	Calinski-Harabasz index	Average distance within clusters	Dunn index
K-means	Euclidean	NA	2	0,5997	54	4838,971	0,1872	1,7036
	Manhattan	NA	2	0,5966	46	4567,604	0,239	1,6803
	Maximum	NA	2	0,6022	60	5155,628	0,1682	1,7209
PAM	Euclidean	NA	2	0,5687	167	4526,721	0,1852	1,5346
	Manhattan	NA	2	0,56	169	4251,8	0,2357	1,4995
	Maximum	NA	2	0,5741	175	4829,643	0,1667	1,5624
DIANA	Euclidean	NA	2	0,6116	0	4754,579	0,1908	1,812
	Manhattan	NA	3	0,5286	157	3150,633	0,2234	1,5326
	Maximum	NA	2	0,6147	0	5093,598	0,1713	1,8411
C-means	Euclidean	NA	2	0,5354	275	4057,71	0,1866	1,431
	Manhattan	NA	2	0,5262	286	3810,203	0,2369	1,3985
	Maximum	NA	2	0,5461	261	4406,152	0,1679	1,4742
Agglomerative clustering	Euclidean	Average	3	0,598	150	1701,038	0,2109	2,0635
		Complete	3	0,5958	157	2045,38	0,2021	1,4552
		Single	2	0,5697	93	6,2857	0,2862	2,3792
		Ward	3	0,4418	211	3980,847	0,1561	0,9495
	Manhattan	Average	2	0,6022	57	4202,272	0,2466	1,8174
		Complete	2	0,5782	221	212,3141	0,3498	2,3777
		Single	2	0,551	96	5,5319	0,3588	2,2504
		Ward	2	0,5848	91	4442,325	0,2383	1,6392
	Maximum	Average	3	0,604	86	2520,29	0,174	1,986
		Complete	3	0,5769	212	1963,469	0,1855	2,0863
		Single	2	0,578	84	6,894	0,2617	2,4875
		Ward	2	0,6186	71	4685,175	0,1768	1,9198

Table 22: Comparison of the validation metrics for the various clustering methods as applied to the downward ramps.

Clustering algorithm	Distance metric	Linkage criterion	Optimal number of clusters	Silhouette coefficient	Number of incorrect cluster assignments	Calinski-Harabasz index	Average distance within clusters	Dunn index
K-means	Euclidean	NA	2	0,5951	33	4613,4	0,186	1,7507
	Manhattan	NA	2	0,587	41	4279,907	0,2367	1,7095
	Maximum	NA	2	0,5956	60	4981,151	0,1667	1,7361
PAM	Euclidean	NA	2	0,5599	158	4355,25	0,1835	1,5668
	Manhattan	NA	2	0,5451	172	4002,734	0,2328	1,5059
	Maximum	NA	2	0,5635	184	4625,914	0,1655	1,5589
DIANA	Euclidean	NA	2	0,6018	0	4573,383	0,1879	1,8361
	Manhattan	NA	2	0,597	0	4281,364	0,2383	1,7816
	Maximum	NA	2	0,6084	0	4969,8	0,1689	1,8633
C-means	Euclidean	NA	2	0,5146	288	3755,075	0,1854	1,4105
	Manhattan	NA	2	0,5014	300	3477,816	0,2348	1,3795
	Maximum	NA	2	0,5278	294	4132,084	0,167	1,4503
Agglomerative clustering	Euclidean	Average	2	0,6197	190	1680,9	0,2353	2,5694
		Complete	2	0,6133	96	3988,117	0,1979	2,0489
		Single	7	0,4709	356	35,9019	0,2712	0,4733
		Ward	3	0,4623	276	3303,057	0,1662	1,2796
	Manhattan	Average	2	0,6154	150	2586,895	0,2734	2,3791
		Complete	6	0,3783	368	2152,739	0,1774	1,2069
		Single	9	0,4456	361	33,2211	0,3375	0,3407
		Ward	2	0,5886	53	4201,567	0,2387	1,7048
	Maximum	Average	2	0,618	161	3091,445	0,1933	2,3634
		Complete	5	0,4097	360	2229,221	0,1416	1,4512
		Single	7	0,4347	360	26,6624	0,2498	0,4094
		Ward	3	0,3915	468	3657,718	0,1427	0,9731

5.5 Summary statistics

This section presents summary statistics of the clusters obtained for the upward and downward ramps, respectively, with the DIANA clustering algorithm using the maximum distance measure and the agglomerative clustering algorithm using the Euclidean distance measure and average linkage criterion. The summary statistics provides insights into the distribution of the data which can be used to characterise a wind energy facility site.

It is possible to compute the distribution of each variable representing the upward or downward ramps by cluster. Each upward ramp and each downward ramp are represented by two variables, namely the

ramp magnitude and the ramp duration. The upward ramps are grouped into 2 clusters using the DIANA algorithm with the maximum distance measure, and the downward ramps are grouped into 2 clusters using the agglomerative algorithm with the Euclidean distance measure and average linkage criterion. Fig. 208 shows the distribution of the ramp magnitude of the upward ramps assigned to cluster 1 and cluster 2, respectively. Fig. 209 shows the distribution of the ramp magnitude of the downward ramps assigned to cluster 1 and cluster 2, respectively. Fig. 210 shows the distribution of the ramp duration of the upward ramps assigned to cluster 1 and 2, respectively. Fig. 211 shows the distribution of the ramp duration of the downward ramps assigned to cluster 1 and 2, respectively.

For the upward ramps, the mean value of the ramp magnitude of the upward ramps assigned to cluster 1 and cluster 2, respectively, is 31.0 MW and 70.8 MW. The mean value of the ramp duration of the upward ramps assigned to cluster 1 and cluster 2, respectively, is 60.1 min and 113.0 min. For the downward ramps, the mean value of the ramp magnitude of the downward ramps assigned to cluster 1 and cluster 2, respectively, is 35.7 MW and 89.8 MW. The mean value of the ramp duration of the downward ramps assigned to cluster 1 and cluster 2, respectively, is 69.8 min and 140 min. The mean magnitude of the clusters for the upward ramps is lower than the mean magnitude of the clusters for the downward ramps. The mean duration of the clusters for the upward ramps is lower than the mean duration of the clusters for the downward ramps. The clusters for the upward ramps, therefore, display slightly different characteristics than the clusters for the downward ramps.

Clustering can be used to characterise a wind energy facility site in terms of ramping mode by employing summary statistics of the obtained clusters. This enables to comparison of two wind energy facility sites based on ramping mode.

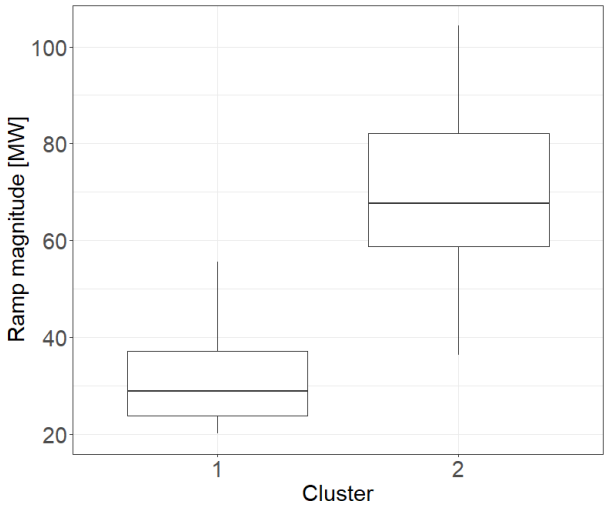


Fig. 208: Boxplot for the ramp magnitude of the upward ramps, detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$, assigned to cluster 1 and cluster 2, respectively, via the DIANA algorithm with the maximum distance measure.

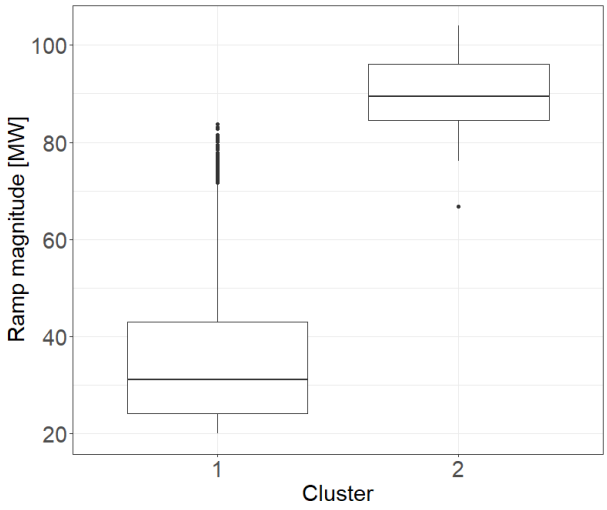


Fig. 209: Boxplot for the ramp magnitude of the downward ramps, detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$, assigned to cluster 1 and cluster 2, respectively, via the agglomerative algorithm with the Euclidean distance measure and average linkage criterion.

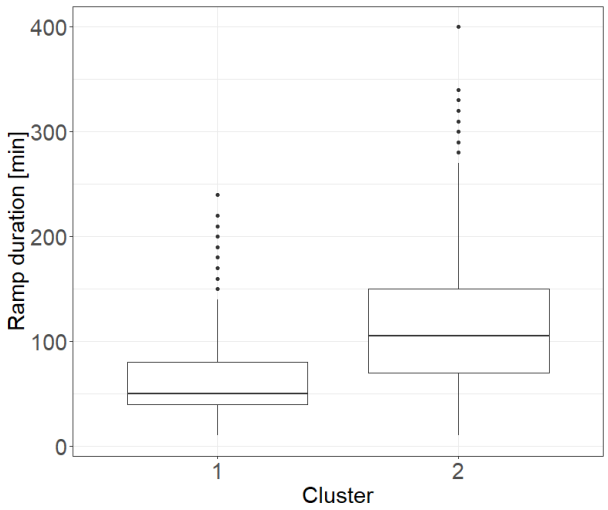


Fig. 210: Boxplot for the ramp duration of the upward ramps, detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$, assigned to cluster 1, respectively, via the DIANA algorithm with the maximum distance measure.

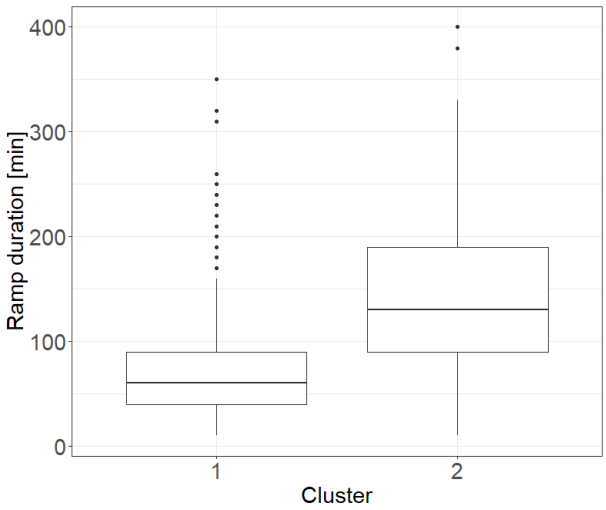


Fig. 211: Boxplot for the ramp duration of the downward ramps, detected by the multi-parameter segmentation algorithm for $\gamma = 0.01,\ \lambda = 5$, assigned to cluster 2, respectively, via the agglomerative algorithm with the Euclidean distance measure and average linkage criterion.

6 Conclusions and Recommendations

6.1 Overview

This section presents the conclusions of the research, as well as the recommendations for future work. The conclusions of the work are presented in light of the project objectives, namely:

- Review the existing models to describe and detect ramp events.
- Investigate the development of improved ramp models and ramp detection algorithms that are closely aligned with the grid impacts induced by variable wind power. The modelling approach should make allowance for short- and long-term impacts of ramping under conditions of variable penetration of wind power.
- Evaluate and compare the performance of the existing and proposed models and ramp detection algorithms, based on the metrics commonly applied in literature.
- Perform statistical analysis of the key ramping features with the view to obtain insights into wind power ramp events that can be used to make informed scheduling decisions as well as to develop forecasting models for ramping.
- Develop a methodology to characterise existing and potential wind energy facility sites in terms of ramping mode. The proposed methodology should make use of historical data from meteorological measurements masts where available. It is envisaged that the model may be of use in the medium- and long-term planning of wind energy deployment.

6.2 Conclusions

6.2.1 Review existing ramp models and ramp detection algorithms

The important findings from the literature concerning the existing models to describe and detect ramp events are as follows:

- *Ramp definition:* One of the major problems associated with wind power ramp event studies is defining a wind power ramp event. There is limited consensus in literature regarding a set definition of a wind power ramp event. Various formal and informal definitions of wind power ramp events have been proposed in literature [12], [14], [18], [21], [38], [46]. Binary classification is often employed to identify wind power ramp events [35]. The binary classification is based on various ramp definitions [12], [18], [28]. Binary classification, however, has several drawbacks including that different ramp definitions result in the identification of different sets of ramps, as well as the idea that all ramps are similar [21], [35]. Furthermore, the ramp function, which characterises ramp events based on the wavelet transform, is proposed in literature to address several drawbacks related to ramp event characterisation based on a binary classification [13]. The ramp function, however, has not been widely used in literature since it is less intuitive than the binary classification.
- *Ramp detection algorithm:* An extensive range of algorithms for detecting wind power ramps have been proposed in literature. Some of the most commonly used ramp detection algorithms to date is the Swinging Door Algorithm (SDA), L1- ramp detect with Sliding Window (L1-SW) and the Optimised Swinging Door Algorithm (OpSDA). These algorithms have been adopted in literature for a wide range of applications. The benefits of the SDA include its structural and computational efficiency as well as its robustness, regardless of noise. The SDA, however, still has several issues associated with its use, including that there is no consensus on how to determine the optimal value of the tuneable parameter and how the optimal ramp segments for the wind power time series should look [34], [54]. The L1-SW is advantageous because it always finds the largest intervals within

the wind power time series that satisfies the user-specified ramp rules and it also ensures that all ramps satisfying the given rules are detected [21]. Additionally, the method facilitates the use of various ramp rules. The OpSDA was proposed to improve upon the SDA and the L1-SW. It is concluded in literature that the OpSDA performs significantly better compared to the original SDA, as well as similarly or better when compared to the L1-SW, while also being more computationally inexpensive. The performance of the various ramp detection algorithms are evaluated and compared in literature based on visual inspection, a comparison with manually detected ramps as well as a suite of metrics which assesses the ramp detection performance based on the detection accuracy of the start- and end-points of the ramps. It is recognised that there is no established standard for ramp detection, as revealed by the lack of agreement on the ramp definition and the shortage of metrics to evaluate the ramp detection performance.

The binary ramp classification was employed in this study to characterise a ramp event, since it is widely used in literature and due to its intuitive nature. It is, however, still necessary to investigate the application of the ramp function in order to understand its inner workings as well as to determine its benefits and limitations, especially compared to the binary ramp classification.

The SDA, L1-SW and OpSDA each has its own favourable attributes that makes its use preferable. Accordingly, there is a trade-off between computational expedience and optimal ramp detection performance. The development of optimal ramp detection algorithm is, therefore, required to improve upon the state-of-the art ramp detection algorithms and to address the trade-off.

The development of an optimal ramp detection algorithm is complicated by the lack of an established standard for ramp detection. It is, therefore, necessary to investigate the development of a standard for ramp detection. More work is consequently required to develop an optimal ramp model, i.e. a ramp model that is closely aligned with the grid impacts induced by variable wind power, as well as metrics to evaluate the performance of ramp detection algorithms. Once a standard is established for ramp detection, the optimal values for the tunable parameters of the respective ramp detection algorithms can be investigated.

6.2.2 The development of improved ramp models and ramp detection algorithms

A new ramp detection algorithm was proposed, namely a multi-parameter segmentation algorithm, based on the ramp event model proposed by Sevlian and Rajagopal [21] as well as the notion that ramp events are significant increases or decreases in wind power over a limited period of time. The aim of the algorithm is to segregate wind power signals into increasing and decreasing ramps to facilitate ramp detection, as well as to ensure that all possible ramps of varying duration are found. The proposed algorithm also includes the application of a post-processing algorithm to discard horizontal segments. After the algorithm is applied to the wind power time series to extract the linear ramps, a user specified definition of a significant ramp is used to identify the wind power ramp events present in the signal. The proposed algorithm has various advantages which include the structural and computational simplicity of the algorithm and its intuitive design. The algorithm also accounts for merging adjacent ramps with the same direction, the management of bumps, and the processing of insignificant ramps, which ensures that the number of ramps used to linearly approximate a signal is minimised.

In order to develop additional ramp detection algorithms, the use of two machine learning methodologies, namely regression analysis and particle swarm optimisation, were investigated. This firstly required a review on the literature pertaining to regression analysis and particle swarm optimisation. It is important

to note that the ramp detection algorithms based on the application of the particle swarm optimisation, as well as the application of the simple linear regression model, are still in its initial stages of development. More work is, therefore, required to produce optimal ramp detection results for these algorithms. The work was included for exploratory purposes.

An adapted version of the multi-parameter segmentation algorithm was consequently developed. Specifically, the application of the post-processing algorithm to discard horizontal segments is replaced by the application of particle swarm optimisation to identify optimal ramps, i.e. sub-intervals that minimises the ramp duration and maximises the ramp magnitude, and in turn also discard insignificant ramps. The employed ramp model is consistent with the informal definition of a ramp event, i.e. a large change in wind power within a short period of time. The proposed algorithm also allows for the consideration of weights that determines the contribution and relative importance of the ramp duration and ramp magnitude in identifying the optimal ramp.

The application of a simple linear regression model was investigated for its potential to extract ramp events from a temporal wind power profile. With a simple linear regression model, it is possible to extract ramps from a signal in a piecewise linear fashion. A threshold parameter is considered for key goodness-of-fit measures of the simple linear regression model, defining the objectives for an adequate model which in turn affects the identified ramps. Several key goodness-of-fit measures of the simple linear regression model were considered as a threshold, which are listed below along with the corresponding benefits and limitations:

- *Coefficient of determination*: Several limitations are associated with thresholding the coefficient of determination. The power generation profiles associated with wind power sources exhibit a relatively high degree of variability and uncertainty, i.e. wind power has a high degree of unexplainable variation. The coefficient of determination of the simple linear regression models estimating the wind power ramps are, therefore, expected to be lower and specifying a value that is too high for the threshold parameter of the coefficient of determination may result in overfitting. Additionally, the degree of variability in the wind power also varies across different ranges of the regressor variable. It is, therefore, not obvious what is considered as a suitable threshold for the coefficient of determination.
- *Standard error of regression*: The advantage of thresholding the regression-based segmentation algorithm with the standard error of regression is that the precision of the model is measured in terms of the units of the response variable, instead of a percentage as for the regression-based segmentation algorithm considering a threshold for the coefficient of determination, thereby, making the model more intuitive.
- *Maximum residual*: The advantage of thresholding the regression-based segmentation algorithm with the maximum residual value is that the precision of the model is measured in terms of the units of the response variable, similar to thresholding the standard error of regression. The regression based segmentation algorithm thresholding the maximum residual, however, ensures that all the residual values of the simple linear regression model representing a ramp is less than or equal to the chosen threshold, whereas thresholding the standard error of regression ensures that the average value of the residuals are less than or equal to the chosen threshold.
- *The slope of the regression line*: The proposed method is able to decompose the temporal wind power signal into upward and downward ramp segments. The drawback of the method is, however, that it does not consider a threshold parameter to allow ramp variations as well as define the sensitivity to ramp variations.

The regression-based segmentation algorithms show a lot of potential, since it allows ramps to be modelled statistically. For the regression-based segmentation algorithms, special attention should be given to the residuals of the simple linear regression model representing the ramps to determine whether the model assumptions are satisfied. It should also be determined how the compliance and non-compliance of the model assumptions affect the detected ramps, and whether it is necessary to satisfy the model assumptions of the simple linear regression model in order to extract adequate ramps. It is necessary to investigate the use of additional properties of the simple linear regression model as a threshold affecting the ramp events. The application of a wide variety of methods estimating the parameters of the simple linear regression model should be investigated to determine which method produces the optimal ramp detection results. Further investigation is required to determine whether outliers in the simple linear regression model may represent ramp variations, and whether thresholding these outliers can define the sensitivity to ramp variations as well as provide information about the location of the ramp end. Furthermore, the ramps for the regression-based segmentation algorithms are anchored to the points of the signal. More work is required to determine whether it is optimal to anchor the ramps to the signal or rather to use the fitted regression line to represent the ramps.

Further investigation is required to determine whether the ramp models employed by the multi-parameter segmentation algorithm, the multi-parameter segmentation algorithm with particle swarm optimisation and the regression-based segmentation algorithm respectively, aligns well with the grid impacts induced by variable wind power and proves to be valuable to the system operator. It is also necessary to determine the optimality of the ramp models.

Additionally, more work is required to determine the optimal value of the tunable parameters for the proposed ramp detection algorithms. As previously mentioned, it is therefore necessary to establish a standard for ramp detection.

6.2.3 Evaluate and compare the performance of the existing and proposed ramp detection algorithms

The existing and proposed ramp detection algorithms are applied to the measured wind power data of a utility size wind farm to evaluate the ramp detection performance. The results of the SDA, L1-SW, OpSDA and the multi-parameter segmentation algorithm were presented for various values of the tunable parameters to investigate the sensitivity of the extracted ramps to its tunable parameters. The multi-parameter segmentation algorithm with particle swarm optimisation and the regression-based segmentation algorithms are still in the initial stages of its development. The results of the multi-parameter segmentation algorithm with particle swarm optimisation and the regression-based segmentation algorithms were, therefore, included only for exploratory purposes. These results, however, still require in-depth investigation, and the ramp detection performance and run-time of these algorithms need to be evaluated and compared with other existing ramp detection algorithms.

The main contribution of the work is the development of the multi-parameter segmentation algorithm. Focus is, therefore, placed on evaluating the performance of the multi-parameter segmentation algorithm by comparing its detection behaviour to that of the SDA, L1-SW and the OpSDA for optimal parameter values. The optimal tunable parameter values chosen for the SDA, OpSDA and L1-SW are consistent with values found in literature, and the optimal tunable parameters for the multi-parameter segmentation algorithm was found by visual inspection. No attempt was made to solve the optimal tunable parameter values for these methods, and it is recognised that the comparison of the significant ramps is dependent on the choice of parameters.

The literature was consulted to obtain relevant metrics and methodologies which could be used to evaluate and compare the ramp detection performance of the multi-parameter segmentation algorithm with that of the SDA, L1-SW and OpSDA. Accordingly, several comparison methodologies were employed which are listed below along with a summary of the corresponding results:

- *Visual inspection of the set of ramps:* The multi-parameter segmentation algorithm successfully identifies significant wind power ramp events. The multi-parameter segmentation algorithm performs superiorly to the SDA, and similarly or better compared to the OpSDA and L1-SW. The ramps detected by the OpSDA and L1-SW are sometimes interjected by a horizontal segment which wrongly extends the ramp duration, while the multi-parameter segmentation algorithm employs a post-processing algorithm to discard the horizontal segments.
- *Comparison of the run-times and number of ramps:* It is recognised that the performance of the SDA and the multi-parameter segmentation algorithm is superior to that of the L1-SW and OpSDA based on run-time. The multi-parameter segmentation algorithm outperforms the SDA for smaller values of epsilon. The multi-parameter segmentation algorithm detects the most ramps among all the methods, and the number of detected ramps is also more robust to parameter changes, when compared to the other methods.
- *The implementation of a mathematical framework to determine equivalent ramps:* The set of ramps detected by the multi-parameter segmentation algorithm is the most similar to the set of ramps detected by the OpSDA.
- *The determination of the detection accuracy of the start- and end-points of the sets of ramps:* The ramp detection performance of the multi-parameter segmentation algorithm is superior to the ramp detection performance of the SDA, OpSDA and the L1-SW based on the detection accuracy of the start- and end-points of the ramps. The multi-parameter segmentation algorithm correctly identifies the start- and end-points of all the detected ramp events.
- *Comparison of the key ramp features:* The duration and magnitude of the ramps detected by the SDA is smaller than the ramps detected by the L1-SW, OpSDA and the multi-parameter segmentation algorithm. The ramps detected by the multi-parameter segmentation algorithm has similar magnitude and duration features as the ramps detected by the OpSDA as well as the L1-SW for smaller values of the tunable parameter. The duration and magnitude of the ramps detected by the L1-SW for larger parameter values is greater than the ramps detected by the L1-SW for smaller values of the tunable parameter, SDA, OpSDA and the multi-parameter segmentation algorithm.

In summary, the proposed algorithm enables the automated identification of significant wind power ramp events. It is concluded that the multi-parameter segmentation algorithm performs significantly better compared to the original SDA, as well as similarly or better when compared to the L1-SW and OpSDA, while also being more computationally inexpensive. The multi-parameter segmentation algorithm can be applied to a variety of cases in power system operations. It can be used to detect ramp events in wind power, solar power, national load profiles, as well as for residual load, i.e. the national load less the renewable component. The multi-parameter segmentation algorithm can be used in forecasting to develop metrics that can specifically evaluate the ramp forecasting performance, which in turn can be used to tune the ramp forecasting model. Additionally, the ramping events detected from historical data by the multi-parameter segmentation algorithm can be used as training data in statistical and machine learning methods to directly predict future wind power ramping events. The successful ramp detection via the multi-parameter segmentation algorithm also enables the analysis of the weather phenomena

causing the wind power ramps, which provides valuable insights into the development of forecasting models.

6.2.4 Statistical analysis of the key ramping features

The wind power ramps, as obtained by the multi-parameter segmentation algorithm for optimal values of the tunable parameters, were used to perform statistical analysis of the key ramp features in order to gain insights into wind power ramp events which can be used in forecasting. The literature was consulted to determine key questions related to key ramp features that were stated and investigated. The statistical analysis provided insights into the distribution and severity of the ramp events, the frequency of occurrence and seasonality of the ramp events and the distribution of the interarrival times of ramps, amongst others. Although the results are specific to a certain ramp definition, wind farm capacity and location, it is still valuable to the system operator.

6.2.5 Cluster analysis to characterise existing and potential wind energy facility sites in terms of ramping mode

The application of cluster analysis to the ramp detection results obtained via the multi-parameter segmentation algorithm for optimal parameter values was investigated to characterise a wind energy facility site in terms of ramping mode. Specifically, clustering is applied to a data set containing the key ramp features of the detected upward ramps, as well as to a data set containing the key ramp features of the detected downward ramps.

A diverse range of clustering algorithms, distance measures and linkage criteria were investigated in this study to determine the optimal clustering procedure for the datasets of interest. The optimal number of clusters for each clustering methodology was found by using the silhouette method and elbow method. The results of the clustering analysis were presented and investigated using several validation metrics.

It is concluded that it is possible to cluster ramp events according to the key ramp features. It was found that the DIANA clustering algorithm using the maximum distance measure provided the optimal clustering results for the upward ramps, and the optimal number of clusters for this clustering methodology is 2. The agglomerative clustering algorithm using the Euclidean distance measure and average linkage criterion provided the optimal clustering results for the downward ramps, and the optimal number of clusters for this clustering methodology is 2. Therefore, it was concluded that the hierarchical clustering algorithms outperform the partitioning and fuzzy c-means clustering methods for the datasets of interest.

Summary statistics are presented for the optimal clustering results, including the distribution of the ramp magnitude and ramp duration of each cluster of the upward and downward ramps, respectively, so to characterise the wind energy facility site. The mean magnitude of the clusters for the upward ramps is lower than the mean magnitude of the clusters for the downward ramps. The mean duration of the clusters for the upward ramps is lower than the mean duration of the clusters for the downward ramps. The clusters for the upward ramps, therefore, display slightly different characteristics than the clusters for the downward ramps. It is, therefore, concluded that clustering can be used to characterise a wind energy facility site in terms of ramping mode by employing summary statistics of the obtained clusters. This enables the comparison of two wind energy facility sites based on ramping mode. Further investigation is required to determine which statistics of the obtained clusters are optimal to characterise the site.

6.3 Recommendations

The recommendations for future work are summarised as follows:

- Investigate the application of the ramp function to understand its inner workings and determine its benefits and limitations, especially compared to binary ramp classification, and determine the optimal methodology to characterise ramp events.
- Further investigate and develop the multi-parameter segmentation algorithm with particle swarm optimisation and the regression-based segmentation algorithms, and evaluate and compare the ramp detection performance of these algorithms with the existing an proposed ramp detection algorithms.
- Investigate the proposed ramp detection algorithms, i.e. the multi-parameter segmentation algorithm, the multi-parameter segmentation algorithm with particle swarm optimisation and the regression-based segmentation algorithms, to determine whether the employed ramp models are closely aligned with the grid impacts induced by variable wind power and is valuable to the system operator.
- Investigate the development of metrics to evaluate the performance of ramp detection algorithms.
- Apply the ramp detection algorithms to different case studies.
- Determine the optimal value of the tunable parameters for the existing and proposed ramp detection algorithms.
- Develop a risk-based short-term forecasting model for ramping for individual wind energy facilities as well as for the aggregated power from the wind energy fleet.
- Investigate which statistics of the obtained clusters are optimal to characterise a wind energy facility site based on ramping mode.

References

[1] G. Ren, J. Liu, J. Wan, Y. Guo, and D. Yu, “Overview of wind power intermittency : Impacts , measurements , and mitigation solutions,” Appl. Energy, vol. 204, pp. 47–65, 2017, doi: 10.1016/j.apenergy.2017.06.098.

[2] M. Bazilian, H. Outhred, A. Miller, and M. Kimble, “Opinion: An energy policy approach to climate change,” Energy Sustain. Dev., vol. 14, no. 4, pp. 253–255, 2010, doi: 10.1016/j.esd.2010.07.007.

[3] G. M. Shafiullah, A. M.t. Oo, A. B. M. Shawkat Ali, and P. Wolfs, “Potential challenges of integrating large-scale wind energy into the power grid-A review,” Renew. Sustain. Energy Rev., vol. 20, pp. 306–321, 2013, doi: 10.1016/j.rser.2012.11.057.

[4] Department of mineral resources and Energy, “Integrated resource plan 2019.”

[5] “Next Generation Wind and Solar Power,” Next Gener. Wind Sol. Power, 2017, doi: 10.1787/9789264268715-en.

[6] ERC; CSIR ; IFPRI, “The developing energy landscape in South Africa: Technical Report,” no. October, pp. 1–42, 2017, [Online]. Available: http://www.erc.uct.ac.za/sites/default/files/image_tool/images/119/Papers-2017/ERC_2017_The_developing_energy_landscape_in_SA.pdf.

[7] Department of Minerals and Energy, “White paper on renewable energy,” Dep. Miner. Energy Repub. South Africa, no. November, 2003, [Online]. Available: https://unfccc.int/files/meetings/seminar/application/pdf/sem_sup1_south_africa.pdf.

[8] DoE, “Integrated resource plan for electricity 2010-2030,” Dep. Energy (Republic South Africa), no. March, p. 73, 2011, doi: 10.1016/j.wneu.2010.05.012.

[9] S. S. Hemmingway, W. Avenue, O. View, and E. London, “Outlining the Renewable Energy Independent Power Producer Procurement Programme (REIPPPP) Empowerment Imperative,” no. August, 2014.

[10] W. B. I. Anton Eberhard, University of Cape Town Joel Kolker and P. I. D. G. James Leigland, “PPIAF: Report No. ACS8826,” pp. 1–64, 2014.

[11] Deloitte, “Renewable energy in South Africa. Valuation insights,” no. November, pp. 1–39, 2019, [Online]. Available: https://www2.deloitte.com/content/dam/Deloitte/za/Documents/finance/za_renewable_energy_south_africa_valuation_insights_march2019.pdf.

[12] B. Greaves, J. Collins, J. Parkes, and A. Tindal, “Temporal forecast uncertainty for ramp events,” Eur. Wind Energy Conf. Exhib. 2009, EWEC 2009, vol. 3, pp. 1985–2008, 2009.

[13] C. Gallego, A. Costa, Á. Cuerva, L. Landberg3, B. Greaves, and J. Collins, “A wavelet-based approach for large wind power ramp characterisation,” Wind Energy, vol. 17, no. April 2013, pp. 657–669, 2014, doi: 10.1002/we.

[14] C. Kamath, “Understanding Wind Ramp Events Through Analysis of Historical Data,” IEEE PES Transm. Distrib. Conf. Expo, 2010.

[15] T. Ackermann, Wind Power in Power Systems. 2005.

[16] A. Florita, B. M. Hodge, and K. Orwig, “Identifying wind and solar ramping events,” IEEE Green Technol. Conf., no. April, pp. 147–152, 2013, doi: 10.1109/GreenTech.2013.30.

[17] H. Zareipour, D. Huang, and W. Rosehart, “Wind power ramp events classification and forecasting: A data mining approach,” IEEE Power Energy Soc. Gen. Meet., no. May 2016, 2011, doi: 10.1109/PES.2011.6039625.

[18] C. A. Ferreira, J. Gama, L. Moreira-Matias, and A. Botterud, “A survey on wind power ramp forecasting,” no. January, 2011.

[19] J. Zhang, A. Florita, B. Hodge, and J. Freedman, “Ramp Forecasting Performance from Improved Short-Term Wind Power Forecasting Preprint,” no. May, 2014.

[20] M. Cui, J. Zhang, C. Feng, A. R. Florita, Y. Sun, and B. M. Hodge, “Characterizing and analyzing ramping events in wind power, solar power, load, and netload,” Renew. Energy, vol. 111, pp. 227–244, 2017, doi: 10.1016/j.renene.2017.04.005.

[21] R. Sevlian and R. Rajagopal, “Detection and Statistics of Wind Power Ramps,” vol. 28, no. 4, pp. 3610–3620, 2013, doi: 10.1109/TPWRS.2013.2266378.

[22] D. Ganger, J. Zhang, and V. Vittal, “Statistical characterization of wind power ramps via extreme value analysis,” IEEE Trans. Power Syst., vol. 29, no. 6, pp. 3118–3119, 2014, doi: 10.1109/TPWRS.2014.2315491.

[23] North American Electric Reliability Corporation, “Accommodating High Levels of Variable Generation,” North Am. Electr. Reliab. Corp., no. April, p. 104, 2009.

[24] M. Obert and M. Pöller, “Assessing the impact of increasing shares of variable generation on system operations in South Africa,” p. 70, 2017.

[25] J. J. Hargreaves and B. F. Hobbs, “Commitment and dispatch with uncertain wind generation by dynamic programming,” IEEE Trans. Sustain. Energy, vol. 3, no. 4, pp. 724–734, 2012, doi: 10.1109/TSTE.2012.2199526.

[26] P. Sørensen et al., “Power fluctuations from large wind farms,” IEEE Trans. Power Syst., vol. 22, no. 3, pp. 958–965, 2007, doi: 10.1109/TPWRS.2007.901615.

[27] H. Chandler, Harnessing Variable Renewables: A Guide to the Balancing Challenge. 2011.

[28] C. W. Potter, E. Grimit, and B. Nijssen, “Potential benefits of a dedicated probabilistic rapid ramp event forecast tool,” 2009 IEEE/PES Power Syst. Conf. Expo. PSCE 2009, pp. 1–5, 2009, doi: 10.1109/PSCE.2009.4840109.

[29] F. Ueckerdt, R. Brecha, and G. Luderer, “Analyzing major challenges of wind and solar variability in power systems,” Renew. Energy, vol. 81, pp. 1–10, 2015, doi: 10.1016/j.renene.2015.03.002.

[30] J. Zhang, Y. Sun, M. Cui, A. R. Florita, D. Ke, and B.-M. Hodge, “An Optimized Swinging Door Algorithm for Identifying Wind Ramping Events,” IEEE Trans. Sustain. Energy, vol. 7, no. 1, pp. 150–162, 2015, doi: 10.1109/tste.2015.2477244.

[31] A. Bossavy, R. Girard, and G. Kariniotakis, “Forecasting uncertainty related to ramps of wind power production,” Eur. Wind Energy Conf. Exhib. 2010, EWEC 2010, vol. 2, no. April 2010, pp. 1099–1107, 2010.

[32] Q. Chen and K. A. Folly, Wind Power Forecasting, vol. 51, no. 28. 2018.

[33] Y. V. Makarov, C. Loutan, J. Ma, and P. de Mello, “Operational impacts of wind generation on California power systems,” IEEE Trans. Power Syst., vol. 24, no. 2, pp. 1039–1050, 2009, doi: 10.1109/TPWRS.2009.2016364.

[34] J. Zhang, M. Cui, B. M. Hodge, A. Florita, and J. Freedman, “Ramp forecasting performance from improved short-term wind power forecasting over multiple spatial and temporal scales,” Energy, vol. 122, pp. 528–541, 2017, doi: 10.1016/j.energy.2017.01.104.

[35] C. Gallego-Castillo, A. Cuerva-Tejero, and O. Lopez-Garcia, “A review on the recent history of wind power ramp forecasting,” Renew. Sustain. Energy Rev., vol. 52, pp. 1148–1157, 2015, doi: 10.1016/j.rser.2015.07.154.

[36] E. Ela and J. Kemper, “Wind Plant Ramping Behavior Wind Plant Ramping Behavior,” no. December, 2009.

[37] C. Kamath, “Associating weather conditions with ramp events in wind power generation,” 2011 IEEE/PES Power Syst. Conf. Expo. PSCE 2011, pp. 1–8, 2011, doi: 10.1109/PSCE.2011.5772527.

[38] K. T. Bradford, R. L. Carpenter, and B. L. Shaw, “FORECASTING SOUTHERN PLAINS WIND RAMP EVENTS USING THE WRF MODEL AT 3-KM National Weather Center Research Experiences for Undergraduates Program Weather Decision Technologies , Inc .,” Weather, pp. 1–10, 2008.

[39] E. Ela and B. Kirby, “ERCOT Event on February 26 , 2008 : Lessons Learned,” no. July, pp. 1–13, 2008, [Online]. Available: http://www.nrel.gov/docs/fy08osti/43373.pdf.

[40] A. Bossavy, R. Girard, and G. Kariniotakis, “Forecasting ramps of wind power production with numerical weather prediction ensembles,” Wind Energy, no. February 2012, pp. 51–63, 2013, doi: 10.1002/we.

[41] Y. Qi and Y. Liu, “Wind Power Ramping Control Using Competitive Game,” IEEE Trans. Sustain. Energy, vol. 7, no. 4, pp. 1516–1524, 2016, doi: 10.1109/TSTE.2016.2558584.

[42] J. W. Taylor, “Probabilistic forecasting of wind power ramp events using autoregressive logit models,” Eur. J. Oper. Res., vol. 259, no. 2, pp. 703–712, 2017, doi: 10.1016/j.ejor.2016.10.041.

[43] J. W. Zack, “Optimization of Wind Power Production Forecast Performance During Critical Periods for Grid Management,” AWEA Wind. Conf., 2007.

[44] M. Saber Eltohamy, M. Said Abdel Moteleb, H. Talaat, S. Fouad Mekhemer, and W. Omran, “Analyzing Wind Power Ramps for High Penetration of Variable Renewable Generation,” 2019 21st Int. Middle East Power Syst. Conf. MEPCON 2019 - Proc., pp. 768–775, 2019, doi: 10.1109/MEPCON47431.2019.9007951.

[45] M. Cui, C. Feng, Z. Wang, and J. Zhang, “Statistical representation of wind power ramps using a generalized Gaussian mixture model,” IEEE Trans. Sustain. Energy, vol. 9, no. 1, pp. 261–272, 2018, doi: 10.1109/TSTE.2017.2727321.

[46] C. Kamath, “Using Simple Statistical Analysis of Historical Data to Understand Wind Ramp Events,” 2010.

[47] H. Zheng and A. Kusiak, “Prediction of wind farm power ramp rates: A data-mining approach,” J. Sol. Energy Eng. Trans. ASME, vol. 131, no. 3, pp. 0310111–0310118, 2009, doi: 10.1115/1.3142727.

[48] C. Ferreira, J. Gama, V. Miranda, and A. Botterud, “Probabilistic ramp detection and forecasting for wind power prediction,” Reliab. Risk Eval. Wind Integr. Power Syst., pp. 29–44, 2013, doi: 10.1007/978-81-322-0987-4_3.

[49] J. C. Gallego Castillo, “Statistical models for short-term wind power ramp forecasting,” 2013.

[50] N. Cutler, M. Kay, K. Jacka, and T. S. Nielsen, “Detecting, categorizing and forecasting large ramps in wind farm power output using meteorological observations and WPPT,” Wind Energy, vol. 10, no. 5, pp. 453–470, 2007, doi: 10.1002/we.235.

[51] E. H. Bristol, “Swinging door trending. Adaptive trend recording?,” 1990.

[52] T. Ouyang, X. Zha, L. Qin, Y. Xiong, and H. Huang, “Model of selecting prediction window in ramps forecasting,” Renew. Energy, vol. 108, pp. 98–107, 2017, doi: 10.1016/j.renene.2017.02.035.

[53] T. Ouyang, X. Zha, L. Qin, Y. He, and Z. Tang, “Prediction of wind power ramp events based on residual correction,” Renew. Energy, vol. 136, pp. 781–792, 2019, doi: 10.1016/j.renene.2019.01.049.

[54] M. Cui, J. Zhang, A. R. Florita, B. M. Hodge, D. Ke, and Y. Sun, “An optimized swinging door algorithm for identifying wind ramping events,” IEEE Trans. Sustain. Energy, vol. 7, no. 1, pp. 150–162, 2016, doi: 10.1109/TSTE.2015.2477244.

[55] R. Sevlian and R. Rajagopal, “Wind power ramps: Detection and statistics,” IEEE Power Energy Soc. Gen. Meet., pp. 1–8, 2012, doi: 10.1109/PESGM.2012.6344969.

[56] M. Cui, V. Krishnan, B. M. Hodge, and J. Zhang, “A Copula-Based Conditional Probabilistic Forecast Model for Wind Power Ramps,” IEEE Trans. Smart Grid, vol. 10, no. 4, pp. 3870–3882, 2019, doi: 10.1109/TSG.2018.2841932.

[57] M. Cui, J. Zhang, Q. Wang, V. Krishnan, and B. M. Hodge, “A data-driven methodology for probabilistic wind power ramp forecasting,” IEEE Trans. Smart Grid, vol. 10, no. 2, pp. 1326–1338, 2019, doi: 10.1109/TSG.2017.2763827.

[58] D. C. Montgomery, E. A. Peck, and G. G. Vining, Introduction to linear regression analysis.

[59] K. A. F. Copeland, Applied Linear Statistical Models, vol. 29, no. 2. 1997.

[60] G. A. F. Seber and A. J. Lee, Linear Regression Analysis. 2003.

[61] S. R. S and S. K. K, “Detection of Outliers in Regression Model for Medical Data,” no. 2, pp. 50–56, 2017.

[62] J. Neter, M. H. Kutner, W. Wasserman, and C. J. Nachtsheim, Applied Linear Regression Models (Third Edition). 1996.

[63] K. A. Marill, “Advanced Statistics : Linear Regression , Part I : Simple Linear Regression,” vol. 11, no. 1, 2004, doi: 10.1197/S1069-6563(03)00600-6.

[64] S. Il Pak and T. H. Oh, “Correlation and simple linear regression,” J. Vet. Clin., vol. 27, no. 4, pp. 427–434, 2010, doi: 10.1007/978-3-319-89993-0_6.

[65] A. H. Seheult, P. J. Green, P. J. Rousseeuw, and A. M. Leroy, Robust Regression and Outlier Detection., vol. 152, no. 1. 1989.

[66] R. Eberhart and James Kennedy, “A New Optimizer Using Particle Swarm Theory,” Sixth Int. Symp. Micro Mach. Hum. Sci., vol. 0-7803–267, pp. 39–43, 1999, doi: 10.1.1.470.3577.

[67] R. Poli, J. Kennedy, and T. Blackwell, “Particle swarm optimization: An overview,” Swarm Intell., vol. 1, no. 1, pp. 33–57, 2007, doi: 10.1007/s11721-007-0002-0.

[68] Y. Shi and R. Eberhart, “Modified particle swarm optimization algorithm for adaptively configuring globally optimal classification and regression trees,” J. Chem. Inf. Model., vol. 49, no. 5, pp. 1144–1153, 2009, doi: 10.1021/ci800374h.

[69] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez, and R. G. Harley, “Particle swarm optimization: Basic concepts, variants and applications in power systems,” IEEE Trans. Evol. Comput., vol. 12, no. 2, pp. 171–195, 2008, doi: 10.1109/TEVC.2007.896686.

[70] D. Wang, D. Tan, and L. Liu, “Particle swarm optimization algorithm: an overview,” Soft Comput., vol. 22, no. 2, pp. 387–408, 2018, doi: 10.1007/s00500-016-2474-6.

[71] R. C. Eberhart and Y. Shi, “Particle swarm optimization: Developments, applications and resources,” Proc. IEEE Conf. Evol. Comput. ICEC, vol. 1, pp. 81–86, 2001, doi: 10.1109/cec.2001.934374.

[72] C. C. Aggarwal, “An Introduction to Cluster Analysis,” Data Clust., pp. 1–28, 2019, doi: 10.1201/9781315373515-1.

[73] G. W. Milligan and M. C. Cooper, “Methodology Review: Clustering Methods,” pp. 329–354.

[74] C. J. Van Vuuren, W. Cape, H. J. Vermeulen, and W. Cape, “Clustering of Wind Resource Data for one of the South African Renewable Energy Development Zones Abstra ct,” pp. 1–28.

[75] A. Kassambara, “Multivariate Analysis 1: Practical Guide To Cluster Analysis in R (real version),” pp. 1–187, 2015, [Online]. Available: https://pe56d.s3.amazonaws.com/o_1b64qp8fdsg81t4k1fongljq24m.pdf?AWSAccessKeyId=AKIAIP3NFJ5OR5FTFOQQ&Signature=ekwWQ2FpwaTjBwVMF20SsNGBSR4%3D&Expires=1522680996&response-content-disposition=inline;filename=%22preview.pdf%22&response-content-type=applicati.

[76] A. Singh, A. Yadav, and A. Rana, “K-means with Three different Distance Metrics,” Int. J. Comput. Appl., vol. 67, no. 10, pp. 13–17, 2013, doi: 10.5120/11430-6785.

[77] L. Kaufman and P. J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis (Wiley Series in Probability and Statistics). 1990.

[78] I. Bin Mohamad and D. Usman, “Standardization and its effects on K-means clustering algorithm,” Res. J. Appl. Sci. Eng. Technol., vol. 6, no. 17, pp. 3299–3303, 2013, doi: 10.19026/rjaset.6.3638.

[79] F. Kovács, C. Legány, and A. Babos, “Cluster Validity Measurement Techniques,” Proc. 6th Int. Symp. Hungarian Res. Comput. Intell., pp. 1–11, 2005.

[80] M. J. Garbade, “Understanding K-means Clustering in Machine Learning,” 2018. https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1 (accessed Aug. 09, 2019).

[81] D. Sisodia, L. Singh, S. Sisodia, and K. Saxena, “Clustering Techniques: A Brief Survey of Different Clustering Algorithms,” Int. J. Latest Trends Eng. Technol., vol. 1, no. 3, pp. 82–87, 2012.

[82] A. Struyf, M. Hubert, and P. Rousseeuw, “Clustering in an Object-Oriented Environment,” J. Stat. Softw., vol. 1, no. 4, 2015, doi: 10.18637/jss.v001.i04.

[83] C. Reddy, “Understanding the concept of Hierarchical clustering Technique,” 2018. https://towardsdatascience.com/understanding-the-concept-of-hierarchical-clustering-technique-c6e8243758ec (accessed Aug. 08, 2019).

[84] A. Stetco, X. J. Zeng, and J. Keane, “Fuzzy C-means++: Fuzzy C-means with effective seeding initialization,” Expert Syst. Appl., vol. 42, no. 21, pp. 7541–7548, 2015, doi: 10.1016/j.eswa.2015.05.014.

[85] B. Kim, J. M. Kim, and G. Yi, “Analysis of clustering evaluation considering features of item response data using data mining technique for setting cut-off scores,” Symmetry (Basel)., vol. 9, no. 5, 2017, doi: 10.3390/sym9050062.

[86] U. Maulik and S. Bandyopadhyay, “Performance Evaluation of Some Clustering Algorithms and Validity Indices æ,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 12, pp. 1650–1654, 2002, doi: 10.1109/TPAMI.2002.1114856.

[87] J. Baarsch and M. E. Celebi, “Investigation of internal validity measures for K-means clustering,” Lect. Notes Eng. Comput. Sci., vol. 2195, pp. 471–476, 2012.

Appendix A Detection accuracy of the ramp start- and end-points

Table 23 to Table 27 show the contingency tables listing the number of wind power ramp events with accurate ramp start, inaccurate ramp start, accurate ramp end and inaccurate ramp end identified by the SDA, OpSDA, L1-SW (based on evaluating the metrics according to both the actual signal and L1-trend filtered signal) and the multi-parameter segmentation algorithm, respectively, for optimal parameter values. Additionally, the corresponding percentage out of the total number of wind power ramp events is also indicated

Table 23 shows the contingency table listing the number of wind power ramp events with accurate ramp start, inaccurate ramp start, accurate ramp end and inaccurate ramp end identified by the SDA for $\varepsilon = 9 \times 10^{-3}$. The number of ramps with an accurate start- and end-point, i.e. start-Yes-end-Yes (sYeY), is 359, which constitutes 16.95% of the total WPREs. The number of detected ramps with an accurate start-point but an inaccurate end-point, i.e. start-Yes-end-No (sYeN), is 575, which comprises 27.15% of the total WPREs. The number of detected ramps with an inaccurate start-point but an accurate end-point, i.e. start-No-end-Yes (sNeY), is 332, which comprises 15.68% of the total detected WPREs, and the number of detected ramps with an inaccurate start- and end-point, i.e. start-No-end-No (sNeN), is 1427, which constitutes 40.23% of the total WPREs. Therefore, majority of the detected ramps have inaccurate start- and end-points.

Table 23: Contingency table based on the start- and end-points of the ramps detected via the SDA with $\varepsilon = 0.009$.

	End (Yes)	**End (No)**	**Total**
Start (Yes)	359 (16.95%)	575 (27.15%)	934
Start (No)	332 (15.68%)	852 (40.23%)	1184
Total	691	1427	2118

Table 24 shows the contingency table listing the number of wind power ramp events with accurate ramp start, inaccurate ramp start, accurate ramp end and inaccurate ramp end identified by the OpSDA for $\varepsilon = 9 \times 10^{-3}$. The number of ramps with an accurate start- and end-point is 2554, which constitutes 43.81% of the total WPREs. The number of detected ramps with an accurate start-point but an inaccurate end-point is 1303, which comprises 22.35% of the total WPREs. The number of detected ramps with an inaccurate start-point but an accurate end-point is 1380, which comprises 23.67% of the total WPREs. The number of detected ramps with an inaccurate start- and end-point is 593, which constitutes 10.17% of the total WPREs. Therefore, the majority of the ramps have accurate start- and end-points.

Table 24: Contingency table based on the start- and end-points of the ramps detected via the OpSDA for $\varepsilon = 0.009$.

	End (Yes)	**End (No)**	**Total**
Start (Yes)	2554 (43.81%)	1303 (22.35%)	3857
Start (No)	1380 (23.67%)	593 (10.17%)	1973
Total	3934	1896	5830

Table 25 shows the contingency table listing the number of wind power ramp events with accurate ramp start, inaccurate ramp start, accurate ramp end and inaccurate ramp end identified by the L1-SW for $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$ as well as the L1-SW for $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$. For the L1-SW with $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$, the number of ramps with an accurate start- and end-point is 616, which constitutes 18.75% of the total WPREs. The number of detected ramps with an accurate start-point but an inaccurate end-

point is 854, which comprises 26.00% of the total WPREs. The number of detected ramps with an inaccurate start-point but an accurate end-point is 746, which comprises 22.71% of the total WPREs. The number of detected ramps with an inaccurate start- and end-point is 1069, which constitutes 32.54% of the total WPREs. Therefore, the majority of the ramps have inaccurate start- and end-points.

For the L1-SW with $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$, the number of ramps with an accurate start- and end-point is 1411, which constitutes 28.28% of the total WPREs. The number of detected ramps with an accurate start-point but an inaccurate end-point is 1480, which comprises 29.66% of the total WPREs. The number of detected ramps with an inaccurate start-point but an accurate end-point is 1025, which comprises 20.54% of the total WPREs. The number of detected ramps with an inaccurate start- and end-point is 1074, which constitutes 21.52% of the total WPREs. Therefore, the majority of the ramps have inaccurate start- and end-points.

Table 25: Contingency table based on the start- and end-points of the ramps detected via the L1-SW with $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$ and $\lambda = 0.02,\ \gamma = 5\ \times 6$, respectively, based on the actual power signal.

Actual power signal is used to calculate accuracy of detection				
	L1-SW with $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$		**L1-SW with $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$**	
	End (Yes)	**End (No)**	**End (Yes)**	**End (No)**
Start (Yes)	616 (18.75%)	854 (26.00%)	1411 (28.28%)	1480 (29.66%)
Start (No)	746 (22.71%)	1069 (32.54%)	1025 (20.54%)	1074 (21.52%)
Total	3285		4990	

Table 26 shows the contingency table listing the number of wind power ramp events with accurate ramp start, inaccurate ramp start, accurate ramp end and inaccurate ramp end identified by the L1-SW for $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$ and the L1-SW for $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$. The metrics were, however, evaluated based on the L1-trend filtered signal instead of the actual power signal. This information was included to account for the fact that the ramps detected by the L1-SW were identified based on the L1-trend filtered signal and not the actual signal. The L1-trend filter was used to eliminate noise and emphasise appropriate trends in the data, as well as to reduce the number of points that are inputted to the dynamic recursion technique.

For the L1-SW with $\lambda = 0.5,\ \gamma = 1\ \times 10^{-4}$, the number of ramps with an accurate start- and end-point is 3216, which constitutes 97.9% of the total WPREs. The number of detected ramps with an inaccurate start-point but an accurate end-point is 69, which comprises 2.1% of the total WPREs. The number of detected ramps with an accurate start-point but an inaccurate end-point is 0, and the number of detected ramps with an inaccurate start- and end-point is 0.Therefore, the majority of the ramps have accurate start- and end-points.

For the L1-SW with $\lambda = 0.02,\ \gamma = 5\ \times 10^{-6}$, the number of ramps with an accurate start- and end-point is 4818, which constitutes 96.55% of the total WPREs. The number of detected ramps with an inaccurate start-point but an accurate end-point is 172, which comprises 3.45% of the total WPREs. The number of detected ramps with an accurate start-point but an inaccurate end-point is 0, and the number of detected ramps with an inaccurate start and end-point is 0. Therefore, the majority of the ramps have accurate start- and end-points.

Table 26: Contingency table based on the start- and end-points of the ramps detected via the L1-SW with $\lambda = 0.5,\ \gamma = 1 \times 10^{-4}$ and $\lambda = 0.02,\ \gamma = 5 \times 6$, respectively, based on the L1 trend filtered power signal.

L1-trend filtered signal used to calculate accuracy of detection				
	L1-SW with $\lambda = 0.5,\ \gamma = 1 \times 10^{-4}$		**L1-SW with $\lambda = 0.02,\ \gamma = 5 \times 10^{-6}$**	
	End (Yes)	**End (No)**	**End (Yes)**	**End (No)**
Start (Yes)	3216 (97.9%)	0 (0.0%)	4818 (96.55%)	0 (0.0%)
Start (No)	69 (2.1%)	0 (0.0%)	172 (3.45%)	0 (0.0%)
Total	3285		4990	

Table 27 shows the contingency table listing the number of wind power ramp events with accurate ramp start, inaccurate ramp start, accurate ramp end and inaccurate ramp end identified by the multi-parameter segmentation algorithm for $\gamma = 0.01, \varphi = 5$. The number of ramps with an accurate start- and end-point is 6306, which constitutes 100% of the total detected WPREs.

Table 27: Contingency table based on the start- and end-points of the ramps detected via the multi-parameter segmentation algorithm with $\gamma = 0.01, \varphi = 5$.

	End (Yes)	**End (No)**	**Total**
Start (Yes)	6306	0	6306
Start (No)	0	0	0
Total	6306	0	6306

Appendix B Determination of the optimal number of clusters

B.1 Optimal number of clusters for k-means clustering

B.1.1 Upward ramps

Fig. 212 to Fig. 214 plots the total within cluster sum of squares as a function of the number of clusters, as obtained with the k-means algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the upward ramps. From Fig. 212 to Fig. 214, the elbow method suggests that the optimal number of clusters to group the upward ramps with the k-means algorithm is $k = 3$ for the Euclidean, Manhattan and maximum distance measures respectively, since a bend in the curve occurs at $k = 3$. Although further reduction in the total within cluster sum of squares is observed, the change from $k = 3$ to $k = 4$ is insignificant compared to the change from $k = 2$ to $k = 3$.

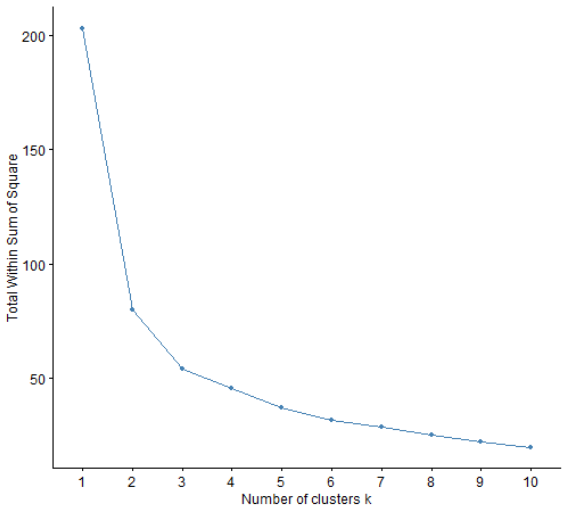


Fig. 212: Total within cluster sum of squares obtained using the k-means algorithm with the Euclidean distance metric when applied to the dataset for the upward ramps.

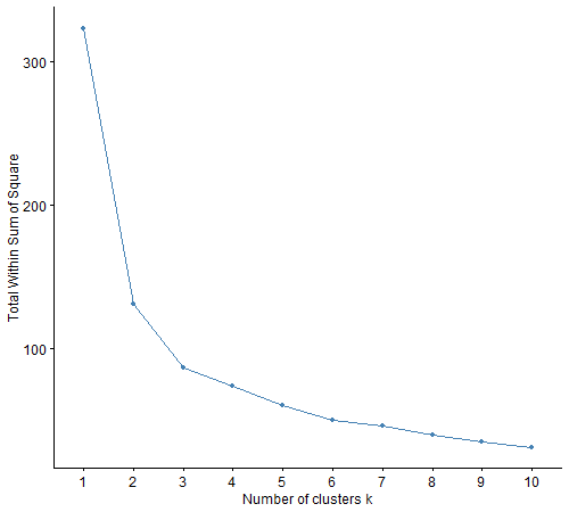


Fig. 213: Total within cluster sum of squares obtained using the k-means algorithm with the Manhattan distance metric when applied to the dataset for the upward ramps.

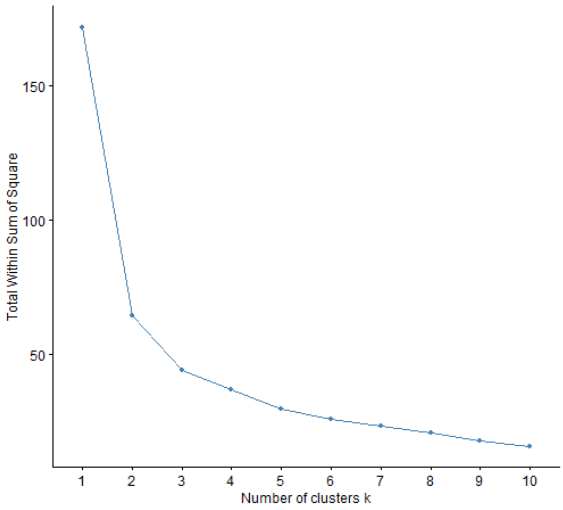


Fig. 214: Total within cluster sum of squares obtained using the k-means algorithm with the maximum distance metric when applied to the dataset for the upward ramps.

Fig. 215 to Fig. 217 plots the average silhouette width as a function of the number of clusters, as obtained with the k-means algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the upward ramps. The silhouette method suggests that the optimal number of clusters to group the upward ramps with the k-means algorithm is $k = 2$ for the Euclidean, Manhattan and maximum distance measures respectively, as shown in Fig. 215 to Fig. 217.

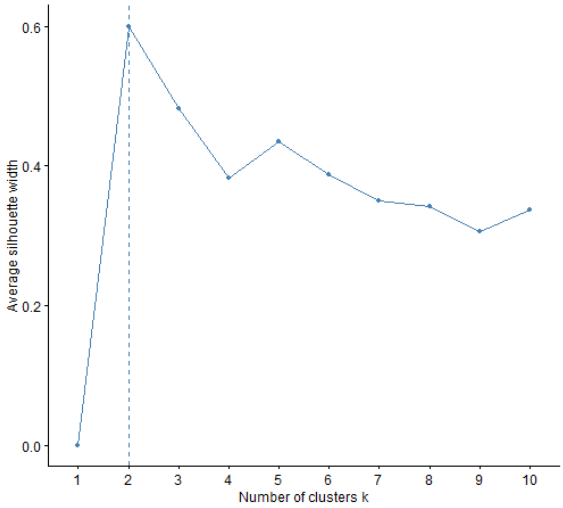


Fig. 215: Average silhouette width obtained using the k-means algorithm with the Euclidean distance metric when applied to the dataset for the upward ramps.

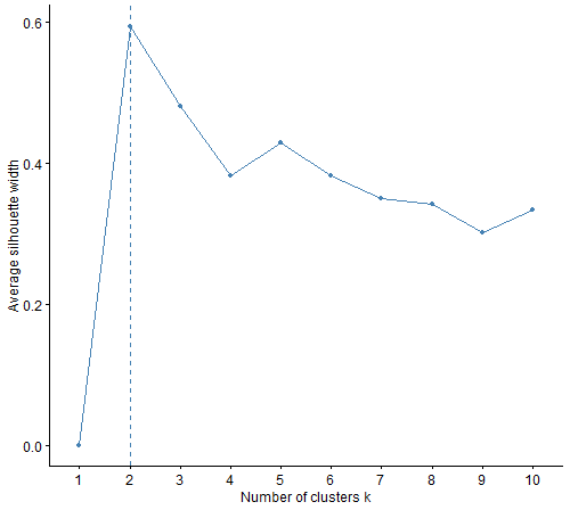


Fig. 216: Average silhouette width obtained using the k-means algorithm with the Manhattan distance metric when applied to the dataset for the upward ramps.

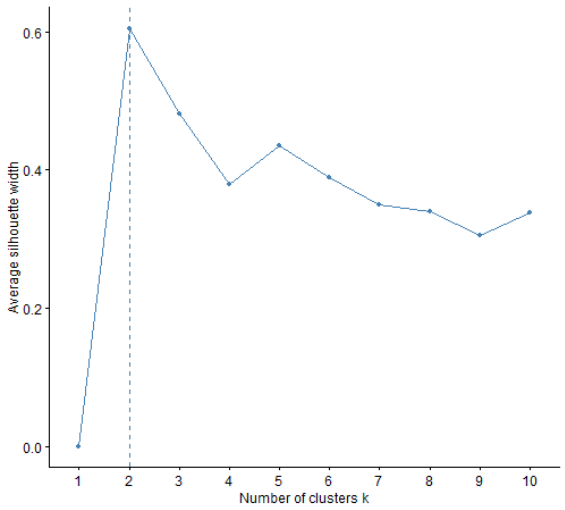


Fig. 217: Average silhouette width obtained using the k-means algorithm with the maximum distance metric when applied to the dataset for the upward ramps.

B.1.2 Downward ramps

Fig. 218 to Fig. 220 plots the total within cluster sum of squares as a function of the number of clusters, as obtained with the k-means algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the downward ramps. From Fig. 218 to Fig. 220, the elbow method suggests that the optimal number of clusters to group the downward ramps with the k-means algorithm is $k = 3$ for the Euclidean, Manhattan and maximum distance measures respectively, since a bend in the curve occurs at $k = 3$. Although further reduction in the total within cluster sum of squares is observed, the change from $k = 3$ to $k = 4$ is insignificant compared to the change from $k = 2$ to $k = 3$.

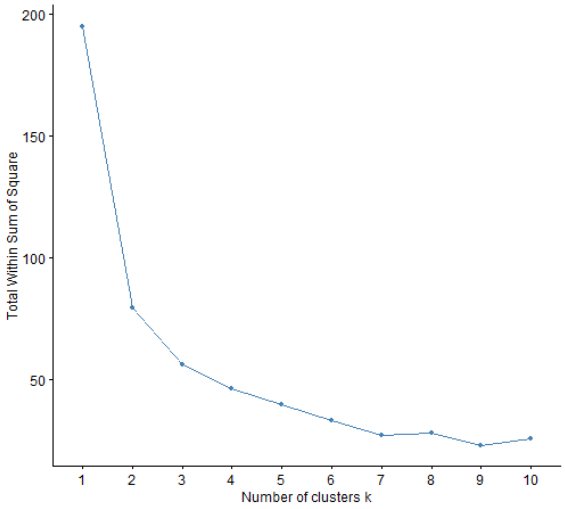


Fig. 218: Total within cluster sum of squares obtained using the k-means algorithm with the Euclidean distance metric when applied to the dataset for the downward ramps.

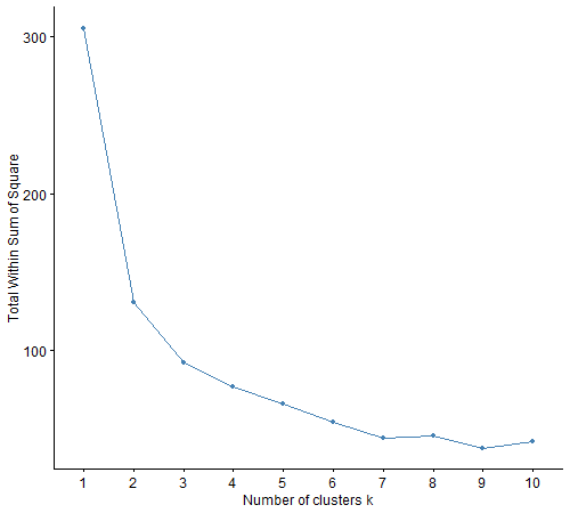


Fig. 219: Total within cluster sum of squares obtained using the k-means algorithm with the Manhattan distance metric when applied to the dataset for the downward ramps.

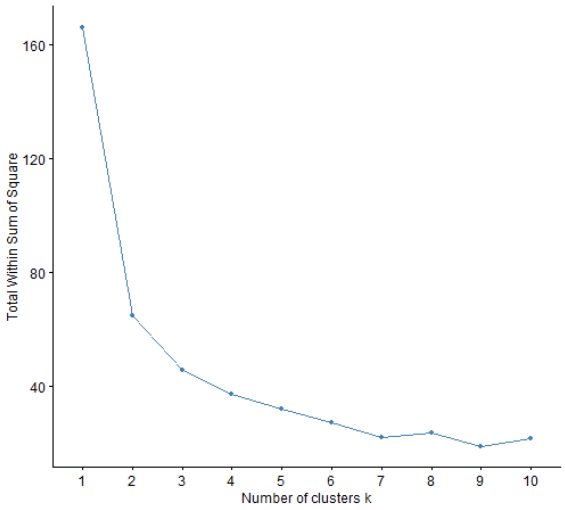


Fig. 220: Total within cluster sum of squares obtained using the k-means algorithm with the maximum distance metric when applied to the dataset for the downward ramps.

Fig. 221 to Fig. 223 plots the average silhouette width as a function of the number of clusters, as obtained with the k-means algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the downward ramps. The silhouette method suggests that the optimal number of clusters to group the downward ramps with the k-means algorithm is $k = 2$ for the Euclidean, Manhattan and maximum distance measures respectively, as shown in Fig. 221 to Fig. 223.

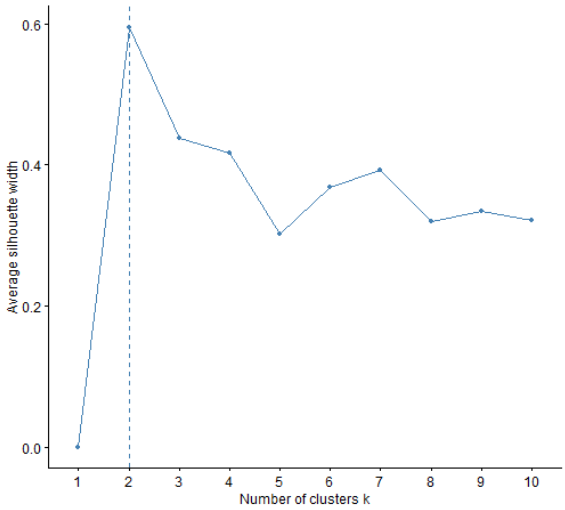


Fig. 221: Average silhouette width obtained using the k-means algorithm with the Euclidean distance metric when applied to the dataset for the downward ramps.

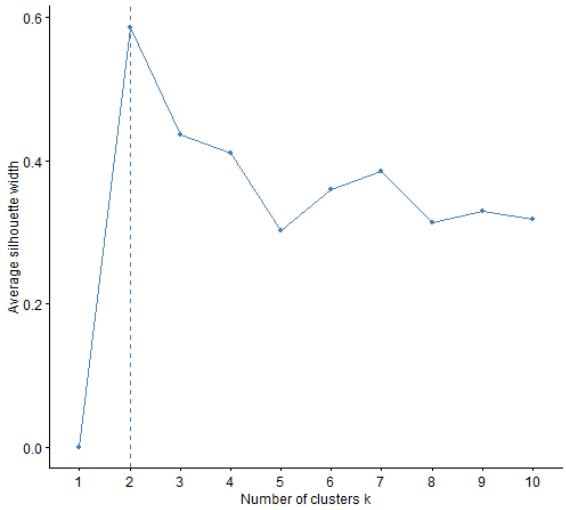


Fig. 222: Average silhouette width obtained using the k-means algorithm with the Manhattan distance metric when applied to the dataset for the downward ramps.

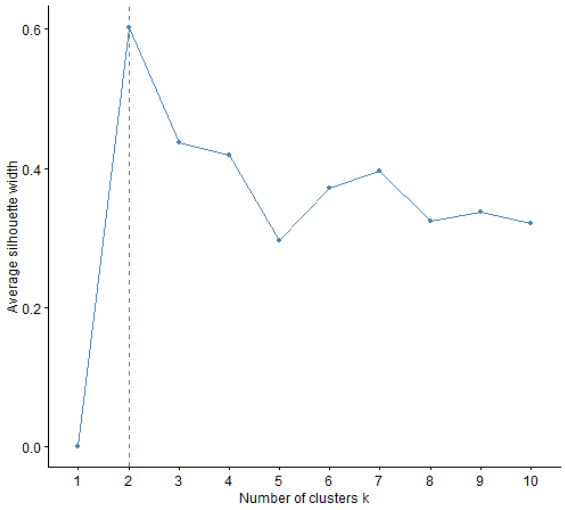


Fig. 223: Average silhouette width obtained using the k-means algorithm with the maximum distance metric when applied to the dataset for the downward ramps.

B.2 Optimal number of clusters for partitioning around medoids clustering

B.2.1 Upward ramps

Fig. 224 to Fig. 226 plots the total within cluster sum of squares as a function of the number of clusters, as obtained with the Partitioning Around Medoids (PAM) algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the upward ramps. From Fig. 224 to Fig. 226, the elbow method suggests that the optimal number of clusters to group the upward ramps with the PAM algorithm is $k = 3$ for the Euclidean, Manhattan and maximum distance measures respectively, since a bend in the curve occurs at $k = 3$. Although further reduction in the total within cluster sum of squares is observed, the change from $k = 3$ to $k = 4$ is insignificant compared to the change from $k = 2$ to $k = 3$.

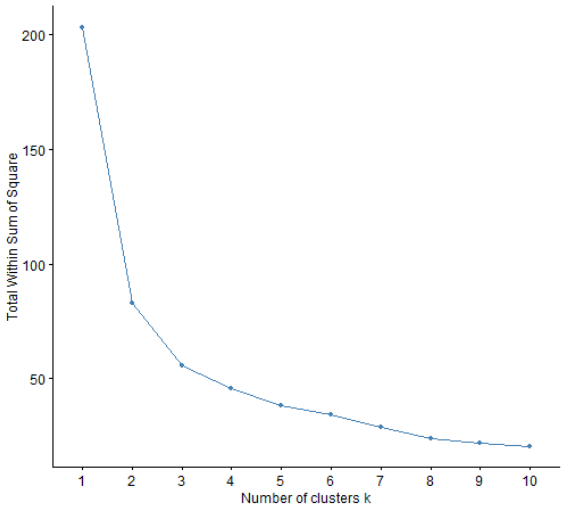


Fig. 224: Total within cluster sum of squares obtained using the PAM algorithm with the Euclidean distance metric when applied to the dataset for the upward ramps.

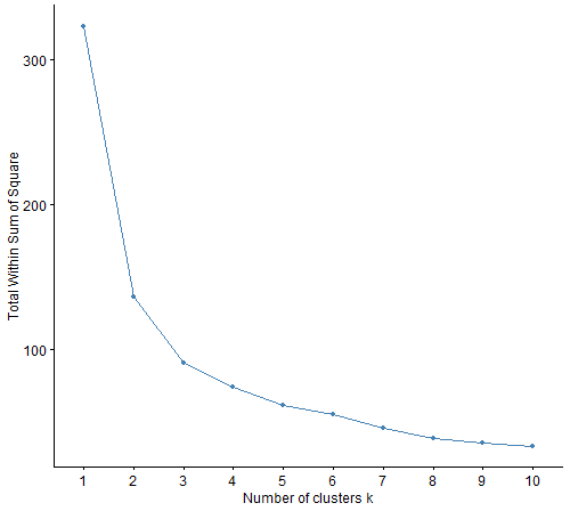


Fig. 225: Total within cluster sum of squares obtained using the PAM algorithm with the Manhattan distance metric when applied to the dataset for the upward ramps.

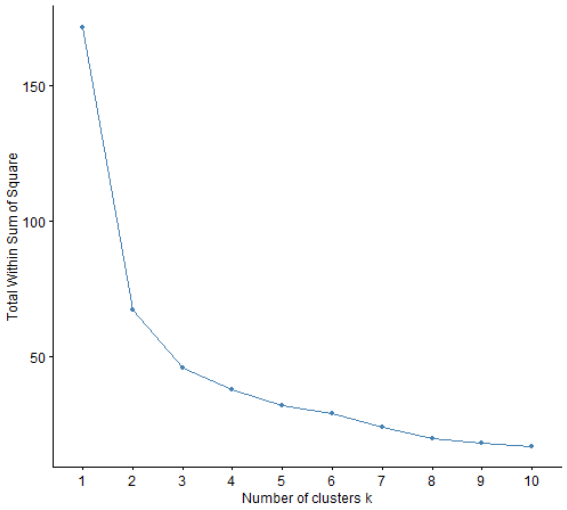


Fig. 226: Total within cluster sum of squares obtained using the PAM algorithm with the maximum distance metric when applied to the dataset for the upward ramps.

Fig. 227 to Fig. 229 plots the average silhouette width as a function of the number of clusters, as obtained with the PAM algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the upward ramps. The silhouette method suggests that the optimal number of clusters to group the upward ramps with the PAM algorithm is $k = 2$ for the Euclidean, Manhattan and maximum distance measures respectively, as shown in Fig. 227 to Fig. 229.

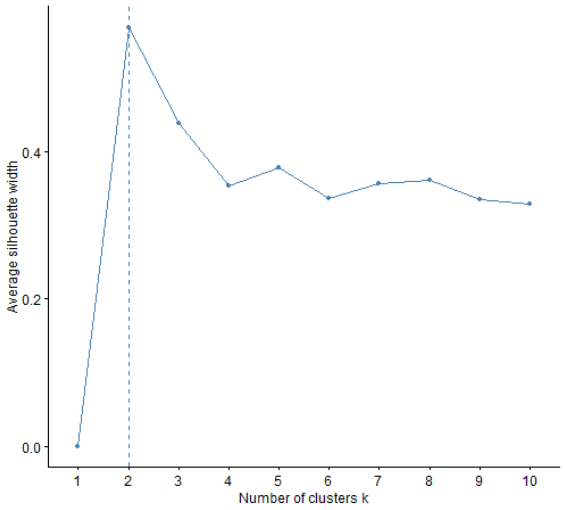


Fig. 227: Average silhouette width obtained using the PAM algorithm and the Euclidean distance metric when applied to the dataset for the upward ramps.

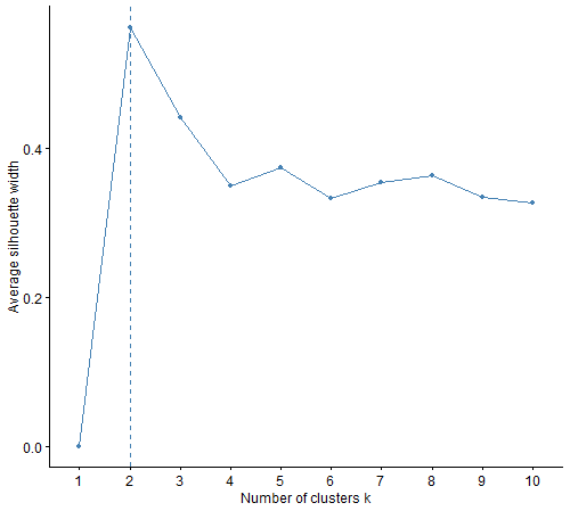


Fig. 228: Average silhouette width obtained using the PAM algorithm and the Manhattan distance metric when applied to the dataset for the upward ramps.

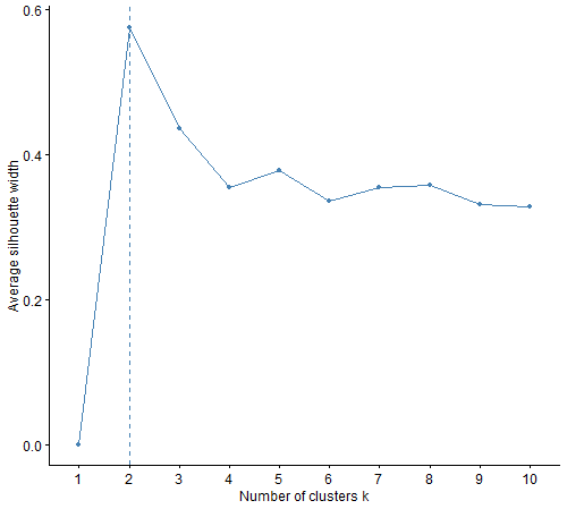


Fig. 229: Average silhouette width obtained using the PAM algorithm and the maximum distance metric when applied to the dataset for the upward ramps.

B.2.2 Downward ramps

Fig. 230 to Fig. 232 plots the total within cluster sum of squares as a function of the number of clusters, as obtained with the PAM algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the downward ramps. From Fig. 230 to Fig. 232, the elbow method suggests that the optimal number of clusters to group the downward ramps with the PAM algorithm is $k = 3$ for the Euclidean, Manhattan and maximum distance measures respectively, since a bend in the curve occurs at $k = 3$. Although further reduction in the total within cluster sum of squares is observed, the change from $k = 3$ to $k = 4$ is insignificant compared to the change from $k = 2$ to $k = 3$.

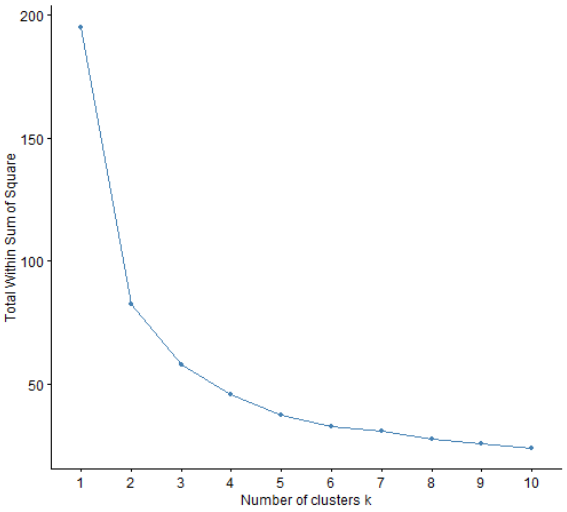


Fig. 230: Total within cluster sum of squares obtained using the PAM algorithm with the Euclidean distance metric when applied to the dataset for the downward ramps.

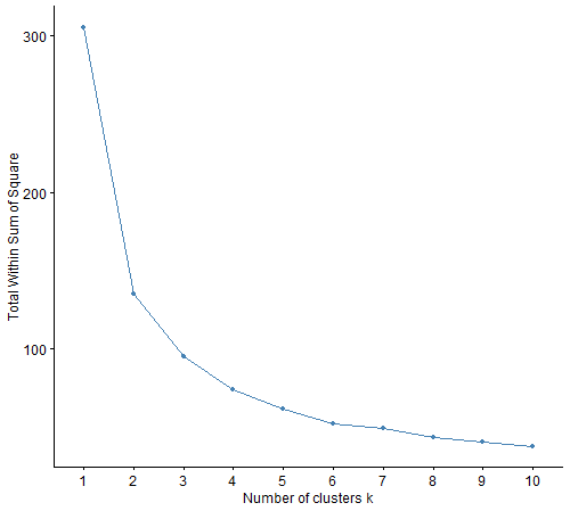


Fig. 231: Total within cluster sum of squares obtained using the PAM algorithm with the Manhattan distance metric when applied to the dataset for the downward ramps.

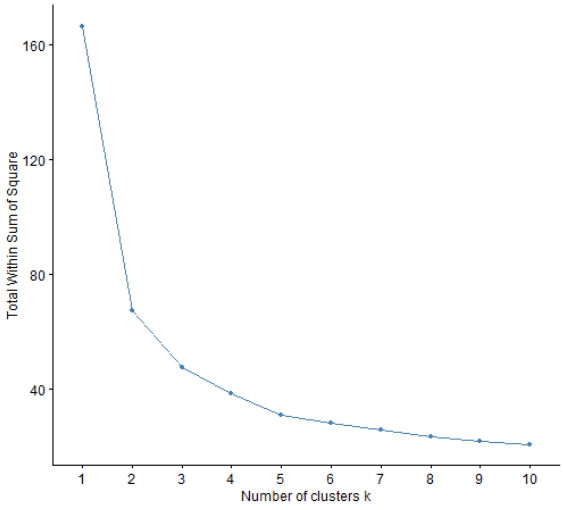


Fig. 232: Total within cluster sum of squares obtained using the PAM algorithm with the maximum distance metric when applied to the dataset for the downward ramps.

Fig. 233 to Fig. 235 plots the average silhouette width as a function of the number of clusters, as obtained with the PAM algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the downward ramps. The silhouette method suggests that the optimal number of clusters to group the downward ramps with the PAM algorithm is $k = 2$ for the Euclidean, Manhattan and maximum distance measures respectively, as shown in Fig. 233 to Fig. 235.

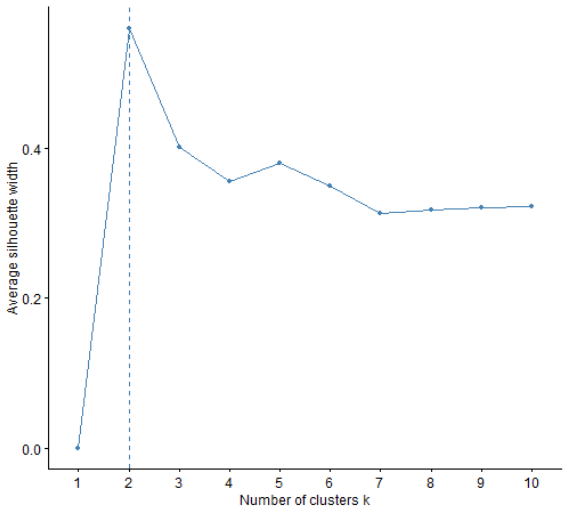


Fig. 233: Average silhouette width obtained using the PAM algorithm with the Euclidean distance metric when applied to the dataset for the downward ramps.

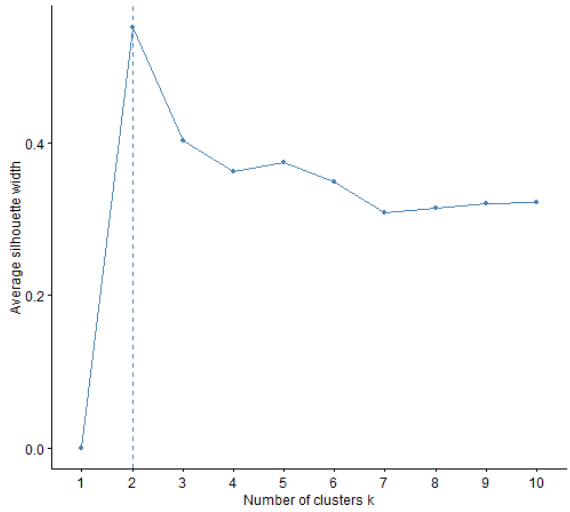


Fig. 234: Average silhouette width obtained using the PAM algorithm with the Manhattan distance metric when applied to the dataset for the downward ramps.

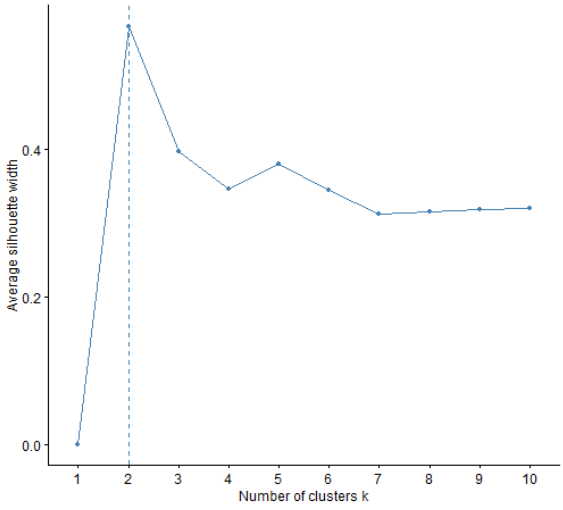


Fig. 235: Average silhouette width obtained using the PAM algorithm with the maximum distance metric when applied to the dataset for the downward ramps.

B.3 Optimal number of clusters for agglomerative nesting clustering

B.3.1 Upward ramps

Fig. 236 to Fig. 239 presents the result of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative clustering algorithm using the Euclidean distance metric and the average, complete, single and Ward linkage criteria, respectively. From Fig. 236 to Fig. 239, the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the average, complete, single and Ward linkage criteria, respectively, is $k = 3, 3, 2$ and 3 according to the majority rule.

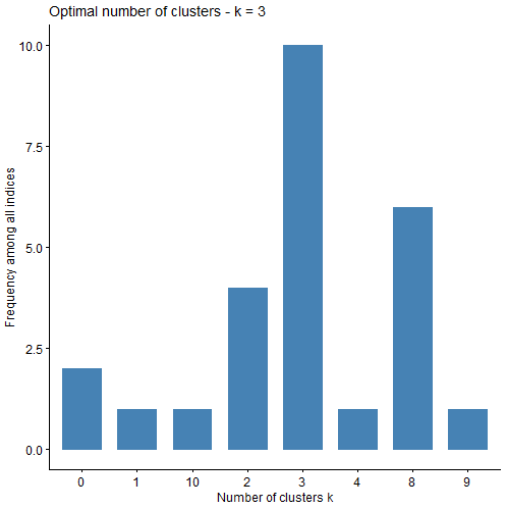


Fig. 236: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the average distance measure.

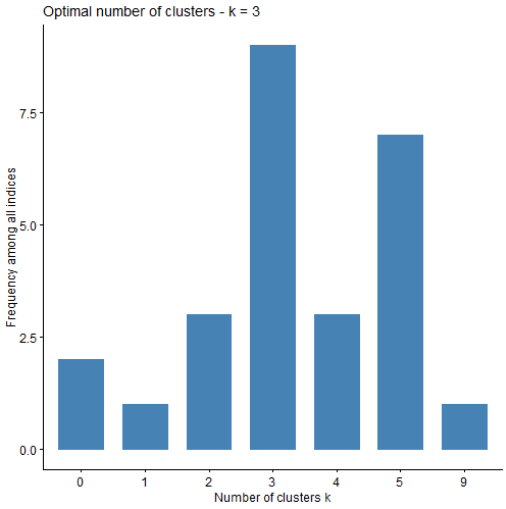


Fig. 237: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the complete distance measure.

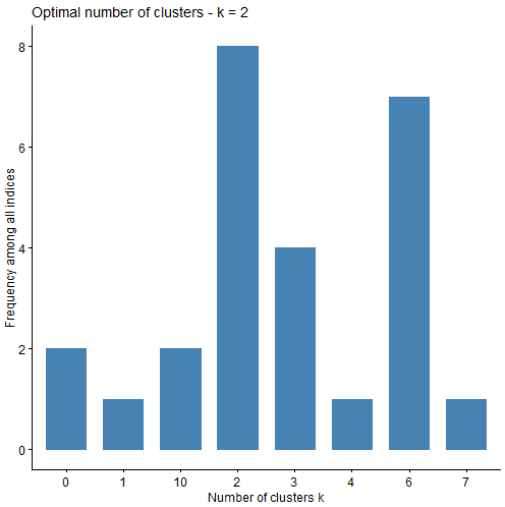


Fig. 238: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the single distance measure.

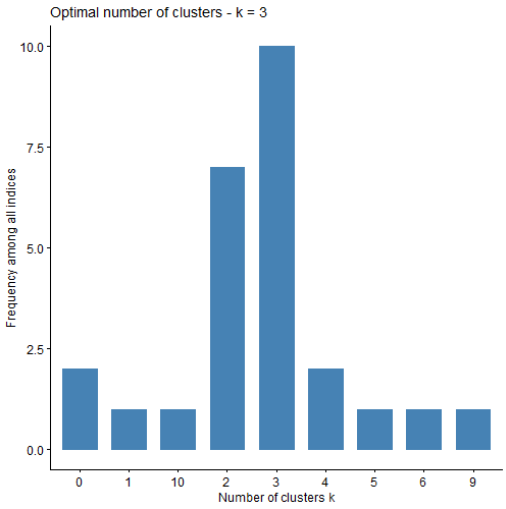


Fig. 239: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Euclidean distance metric and the Ward distance measure.

Fig. 240 to Fig. 243 presents the result of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the average, complete, single and Ward linkage criteria, respectively. From Fig. 240 to Fig. 243, the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the average, complete, single and Ward linkage criteria, respectively, is $k = 2$ according to the majority rule.

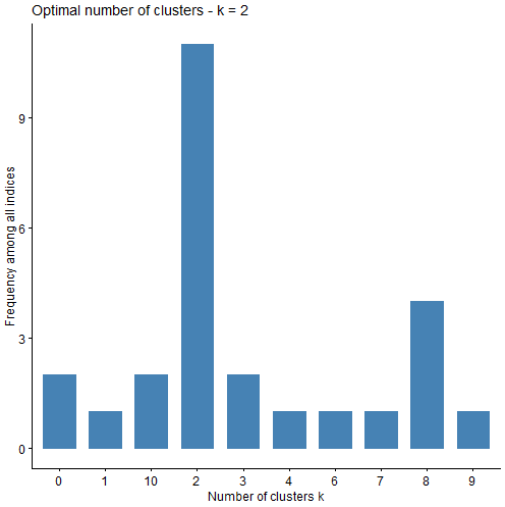


Fig. 240: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the average distance measure.

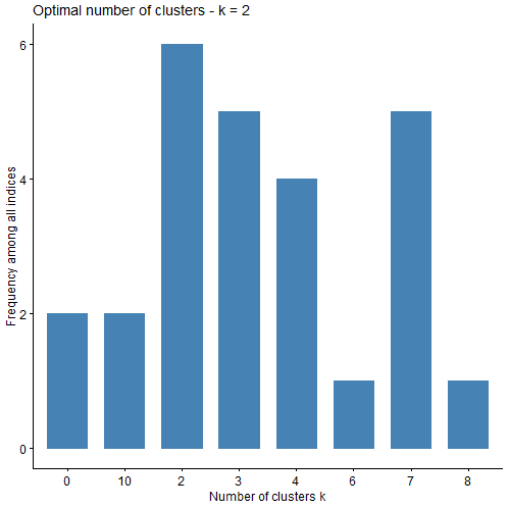


Fig. 241: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the complete distance measure.

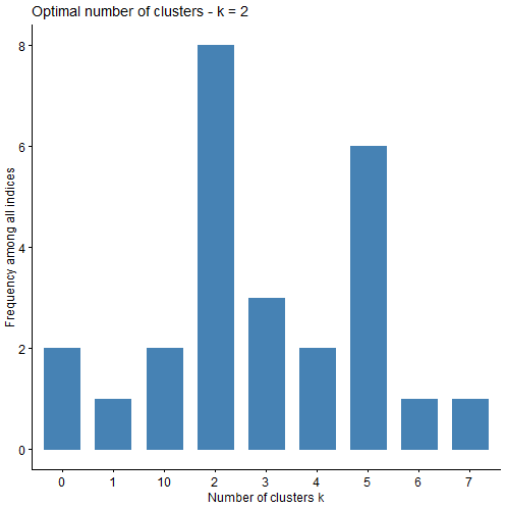


Fig. 242: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the single distance measure.

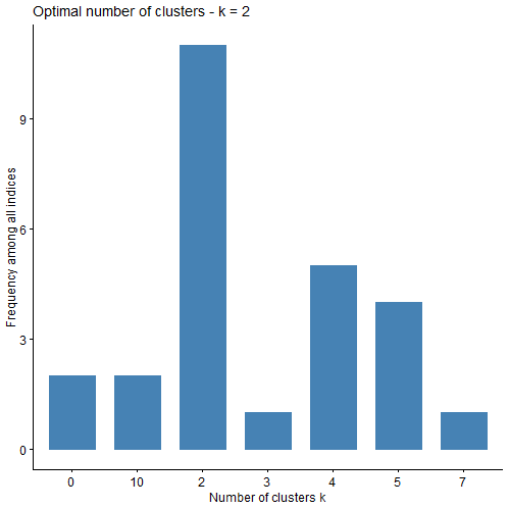


Fig. 243: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative algorithm using the Manhattan distance metric and the Ward distance measure.

Fig. 244 to Fig. 247 presents the result of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative clustering algorithm using the maximum distance metric and the average, complete, single and Ward linkage criteria, respectively. From Fig. 244 to Fig. 247, the optimal number of clusters to group the upward ramps with the agglomerative clustering algorithm using the maximum distance metric and the average, complete, single and Ward linkage criteria, respectively, is $k = 3, 3, 2$ and 2 according to the majority rule.

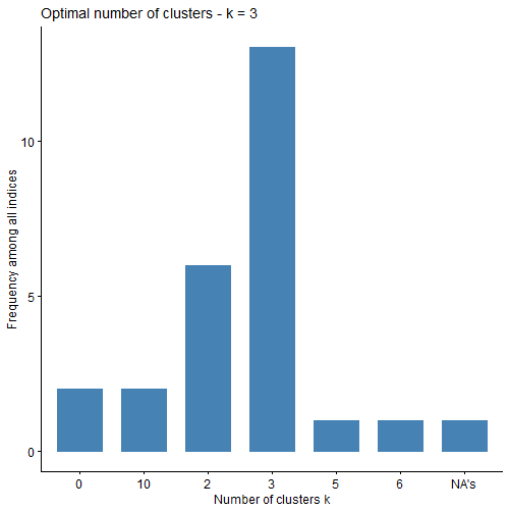


Fig. 244: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative clustering algorithm using the maximum distance metric and the average distance measure.

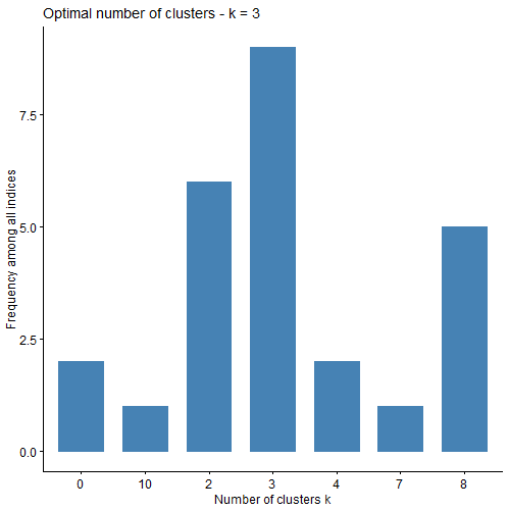


Fig. 245: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative clustering algorithm using the maximum distance metric and the complete distance measure.

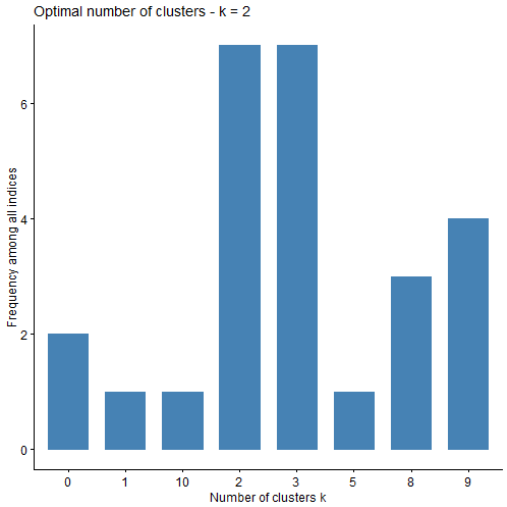


Fig. 246: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative clustering algorithm using the maximum distance metric and the single distance measure.

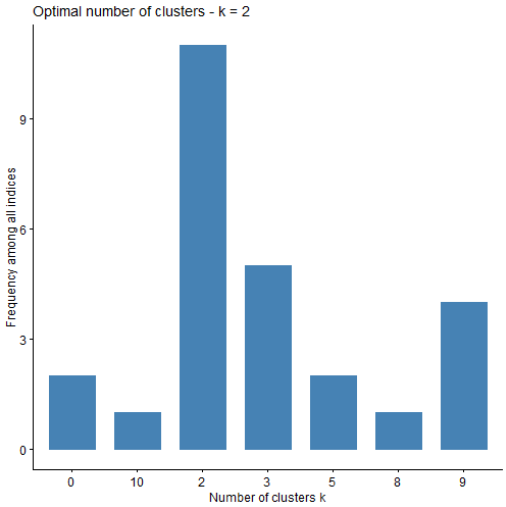


Fig. 247: Results of 30 indices determining the optimal number of clusters to group the upward ramps with the agglomerative clustering algorithm using the maximum distance metric and the Ward distance measure.

B.3.2 Downward ramps

Fig. 248 to Fig. 251 presents the result of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Euclidean distance metric and the average, complete, single and Ward linkage criteria, respectively. From Fig. 248 to Fig. 251, the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Euclidean distance metric and the average, complete, single and Ward linkage criteria, respectively, is $k = 2, 2, 7$ and 3 according to the majority rule.

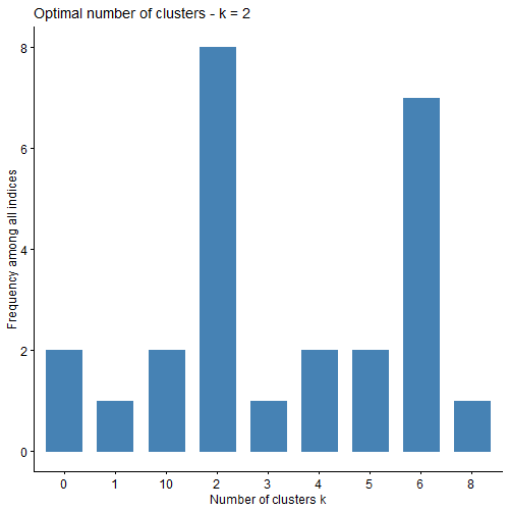


Fig. 248: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Euclidean distance metric and the average distance measure.

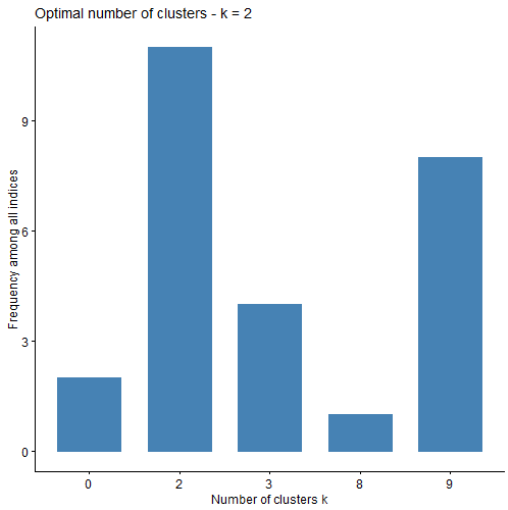


Fig. 249: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Euclidean distance metric and the complete distance measure.

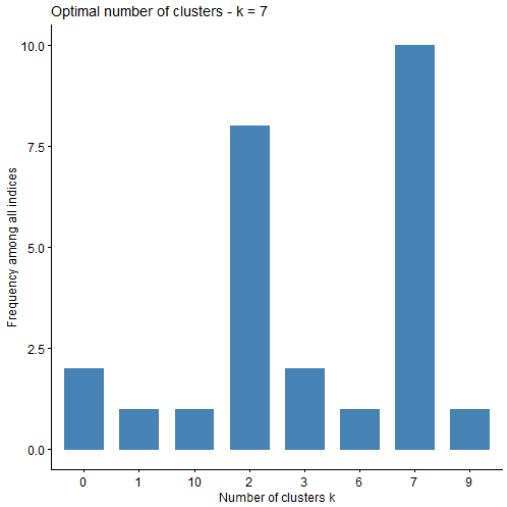


Fig. 250: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Euclidean distance metric and the single distance measure.

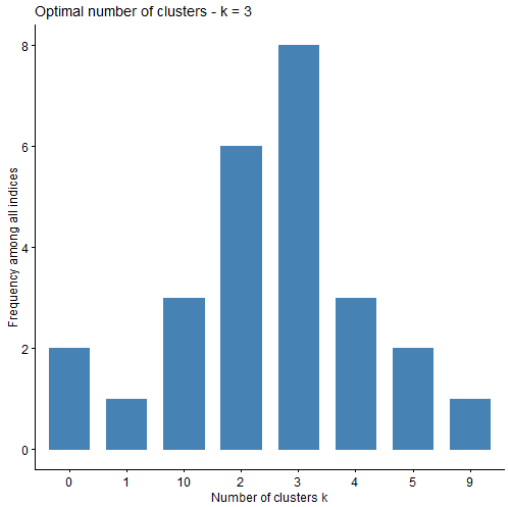


Fig. 251: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative algorithm using the Euclidean distance metric and the Ward distance measure.

Fig. 252 to Fig. 255 presents the result of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Manhattan distance metric and the average, complete, single and Ward linkage criteria, respectively. From Fig. 252 to Fig. 255, the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Manhattan distance metric and the average, complete, single and Ward linkage criteria, respectively, is $k = 2, 6, 9$ and 2 according to the majority rule.

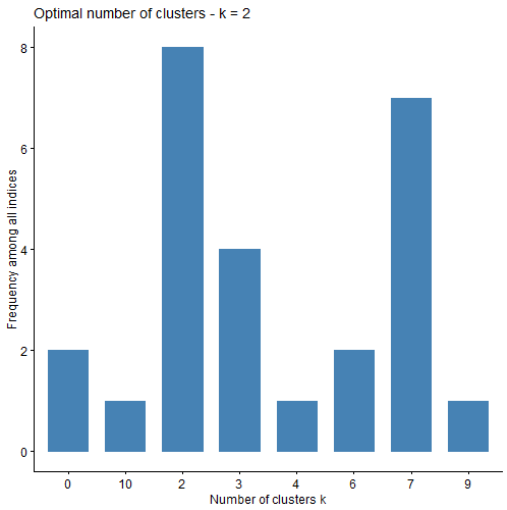


Fig. 252: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Manhattan distance metric and the average distance measure.

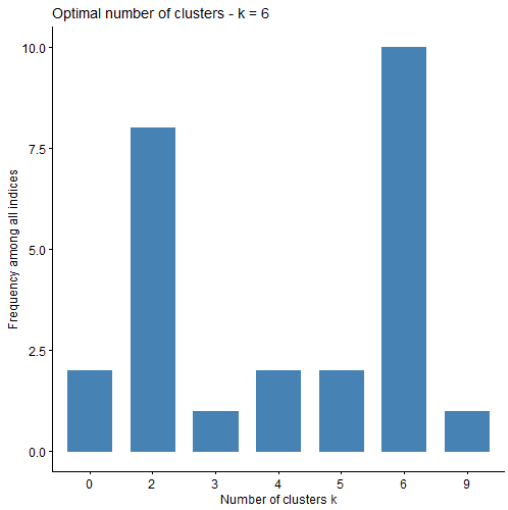


Fig. 253: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Manhattan distance metric and the complete distance measure.

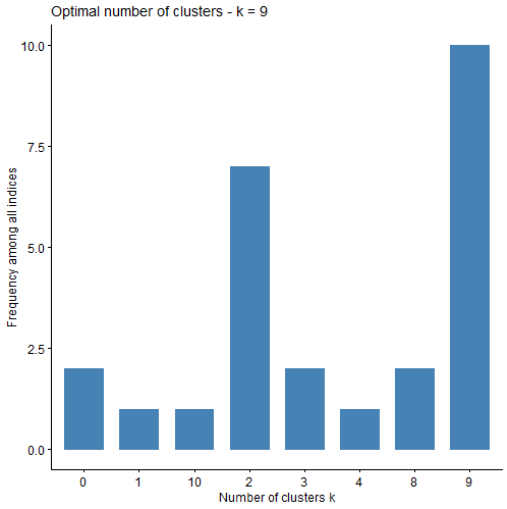


Fig. 254: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Manhattan distance metric and the single distance measure.

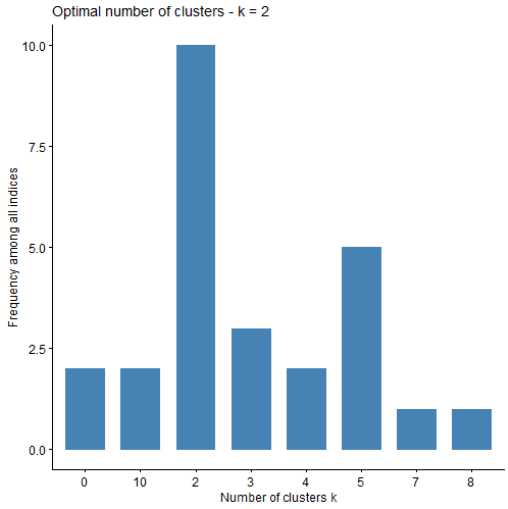


Fig. 255: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the Manhattan distance metric and the Ward distance measure.

Fig. 256 to Fig. 259 presents the result of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the maximum distance metric and the average, complete, single and Ward linkage criteria, respectively. From Fig. 256 to Fig. 259, the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the maximum distance metric and the average, complete, single and Ward linkage criteria, respectively, is $k = 2, 5, 7$ and 3 according to the majority rule.

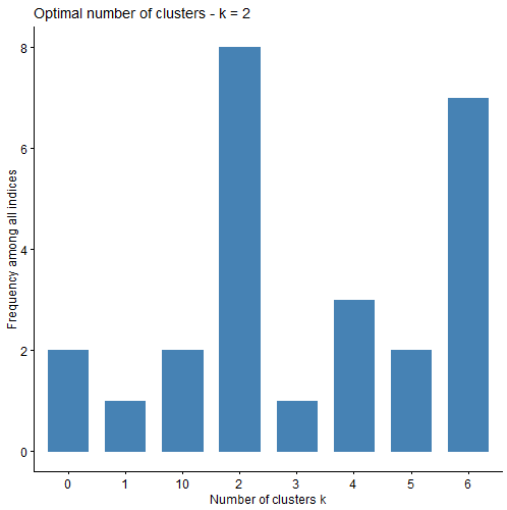


Fig. 256: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the maximum distance metric and the average distance measure.

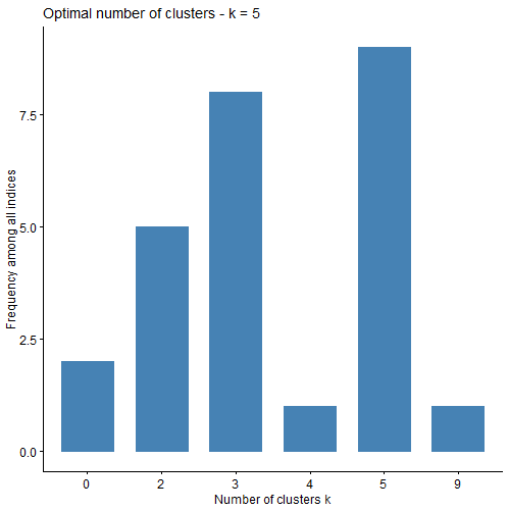


Fig. 257: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the maximum distance metric and the complete distance measure.

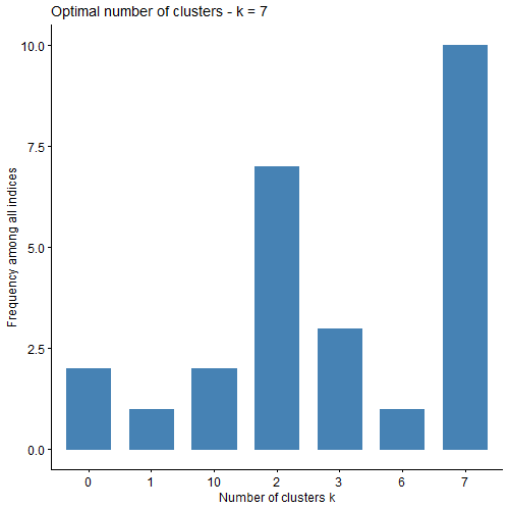


Fig. 258: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the maximum distance metric and the single distance measure.

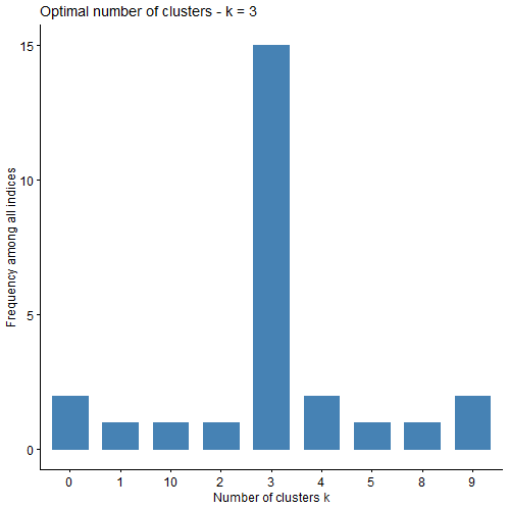


Fig. 259: Results of 30 indices determining the optimal number of clusters to group the downward ramps with the agglomerative clustering algorithm using the maximum distance metric and the Ward distance measure.

B.4 Optimal number of clusters for divisive analysis clustering

B.4.1 Upward ramps

Fig. 260 to Fig. 262 plot the total within cluster sum of squares as a function of the number of clusters, as obtained with the Divisive Analysis (DIANA) clustering algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the upward ramps. From Fig. 260 to Fig. 262, the elbow method suggests that the optimal number of clusters to group the upward ramps with the DIANA clustering algorithm is $k = 3$ for the Euclidean, Manhattan and maximum distance

measures respectively, since a bend in the curve occurs at $k = 3$. Although further reduction in the total within cluster sum of squares is observed, the change from $k = 3$ to $k = 4$ is insignificant compared to the change from $k = 2$ to $k = 3$.

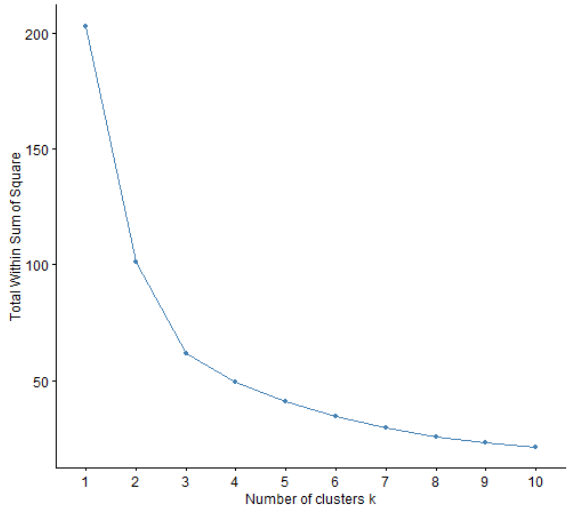


Fig. 260: Total within cluster sum of squares obtained using the DIANA clustering algorithm with the Euclidean distance metric when applied to the dataset for the upward ramps.

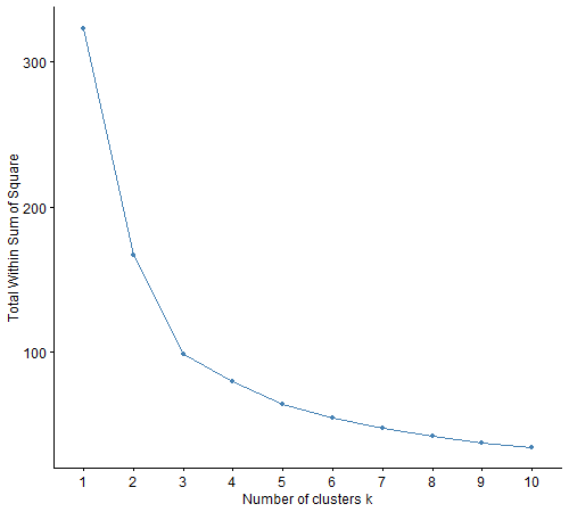


Fig. 261: Total within cluster sum of squares obtained using the DIANA clustering algorithm with the Manhattan distance metric when applied to the dataset for the upward ramps.

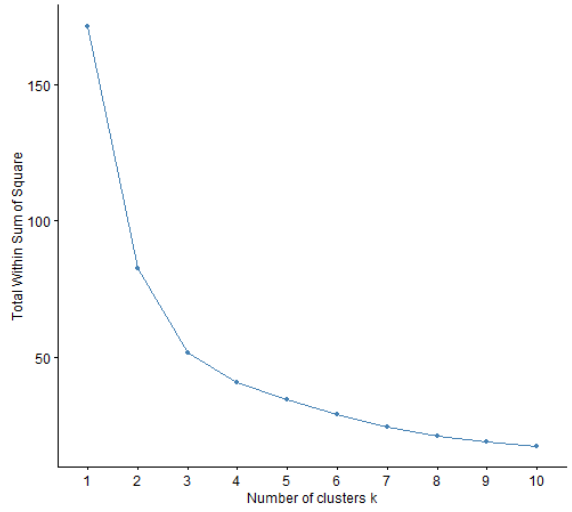


Fig. 262: Total within cluster sum of squares obtained using the DIANA clustering algorithm with the maximum distance metric when applied to the dataset for the upward ramps.

Fig. 263 to Fig. 265 plots the average silhouette width as a function of the number of clusters, as obtained with the DIANA clustering algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the upward ramps. The silhouette method suggests that the optimal number of clusters to group the upward ramps with the DIANA clustering algorithm is $k = 2$ for the Euclidean and maximum distance measures respectively and $k = 3$ for the Manhattan distance measure as shown in Fig. 263 to Fig. 265.

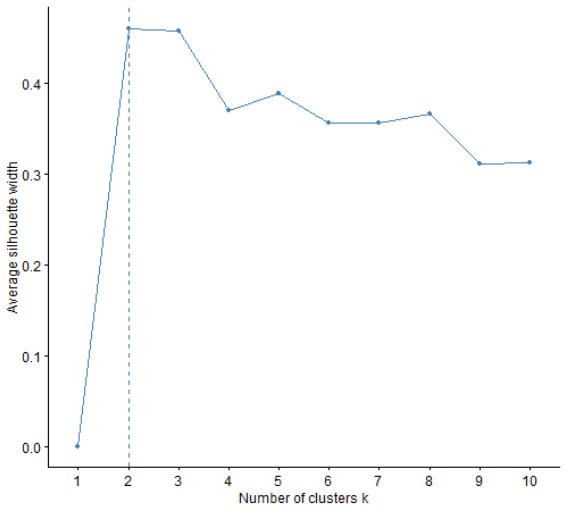


Fig. 263: Average silhouette width obtained using the DIANA algorithm with the Euclidean distance metric when applied to the dataset for the upward ramps.

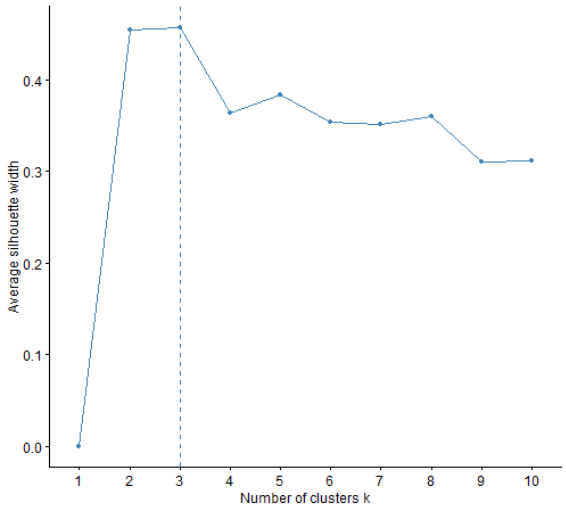


Fig. 264: Average silhouette width obtained using the DIANA algorithm with the Manhattan distance metric when applied to the dataset for the upward ramps.

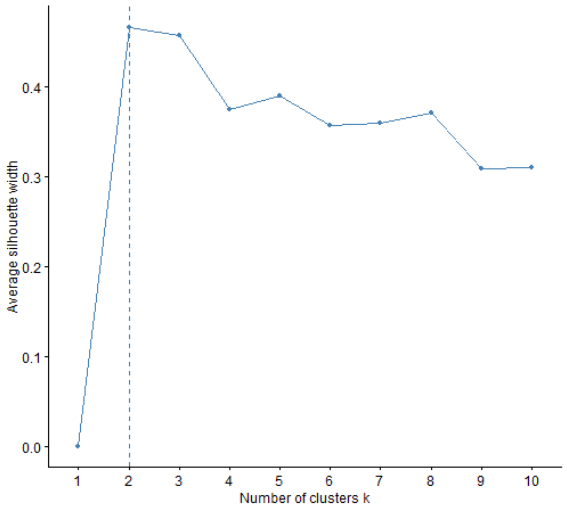


Fig. 265: Average silhouette width obtained using the DIANA algorithm with the maximum distance metric when applied to the dataset for the upward ramps.

B.4.2 Downward ramps

Fig. 266 to Fig. 268 plots the total within cluster sum of squares as a function of the number of clusters, as obtained with the DIANA clustering algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the downward ramps. From Fig. 266 to Fig. 268, the elbow method suggests that the optimal number of clusters to group the downward ramps with the DIANA clustering algorithm is $k = 3$ for the Euclidean, Manhattan and maximum distance measures respectively, since a bend in the curve occurs at $k = 3$. Although further reduction in the total within cluster sum of squares is observed, the change from $k = 3$ to $k = 4$ is insignificant compared to the change from $k = 2$ to $k = 3$.

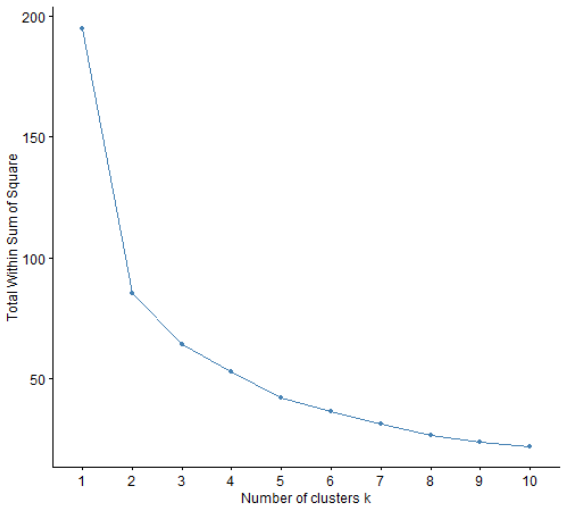


Fig. 266: Total within cluster sum of squares obtained using the DIANA clustering algorithm with the Euclidean distance metric when applied to the dataset for the downward ramps.

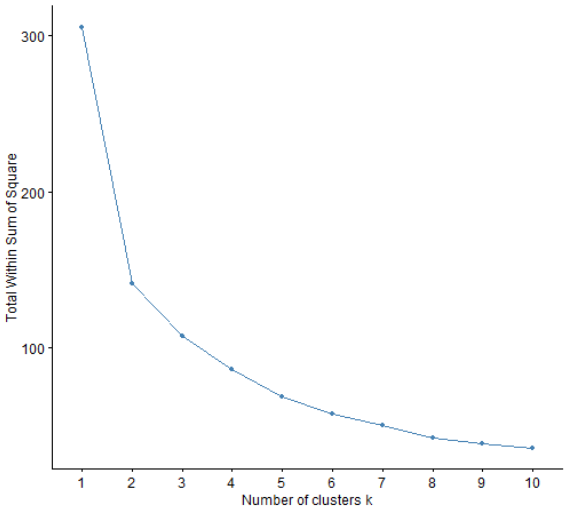


Fig. 267: Total within cluster sum of squares obtained using the DIANA clustering algorithm with the Manhattan distance metric when applied to the dataset for the downward ramps.

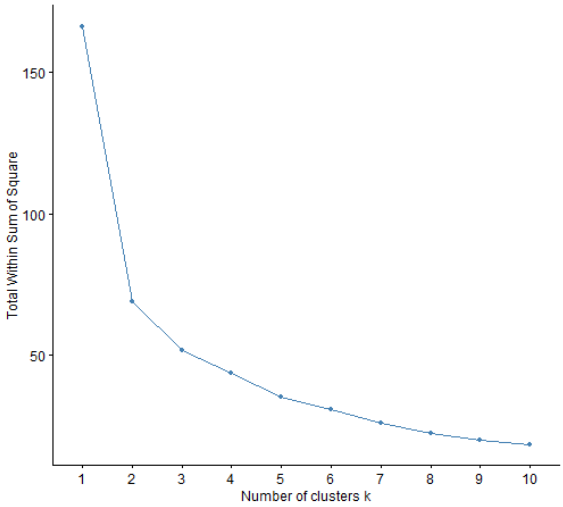


Fig. 268: Total within cluster sum of squares obtained using the DIANA clustering algorithm with the maximum distance metric when applied to the dataset for the downward ramps.

Fig. 269 to Fig. 271 plots the average silhouette width as a function of the number of clusters, as obtained with the DIANA algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the downward ramps. The silhouette method suggests that the optimal number of clusters to group the downward ramps with the DIANA clustering algorithm is $k = 2$ for the Euclidean, Manhattan and maximum distance measures respectively, as shown in Fig. 269 to Fig. 271.

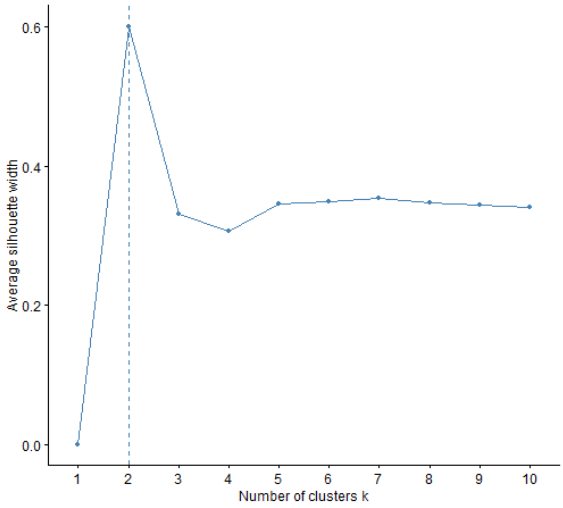


Fig. 269: Average silhouette width obtained using the DIANA clustering algorithm with the Euclidean distance metric when applied to the dataset for the downward ramps.

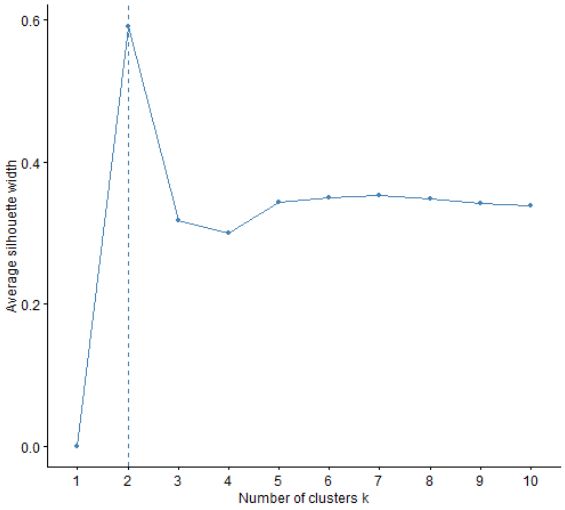


Fig. 270: Average silhouette width obtained using the DIANA clustering algorithm with the Manhattan distance metric when applied to the dataset for the downward ramps.

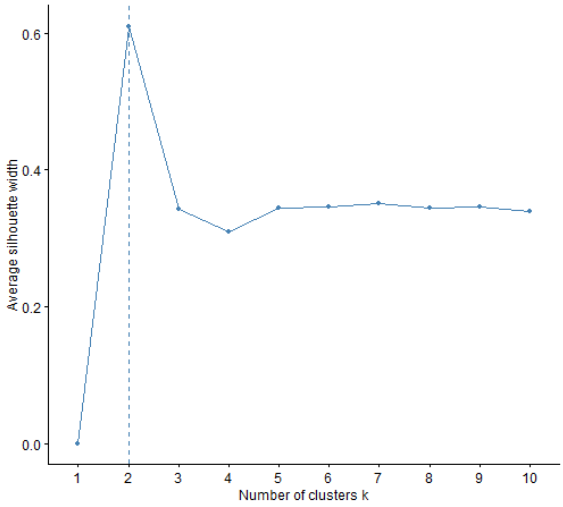


Fig. 271: Average silhouette width obtained using the DIANA algorithm with the maximum distance metric when applied to the dataset for the downward ramps.

B.5 Optimal number of clusters for c-means

B.5.1 Upward ramps

Fig. 272 to Fig. 274 plots the total within cluster sum of squares as a function of the number of clusters, as obtained with the c-means algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the upward ramps. From Fig. 272 to Fig. 274, the elbow method suggests that the optimal number of clusters to group the upward ramps with the c-means algorithm is $k = 3$ for the Euclidean, Manhattan and maximum distance measures respectively, since a

bend in the curve occurs at $k = 3$. Although further reduction in the total within cluster sum of squares is observed, the change from $k = 3$ to $k = 4$ is insignificant compared to the change from $k = 2$ to $k = 3$.

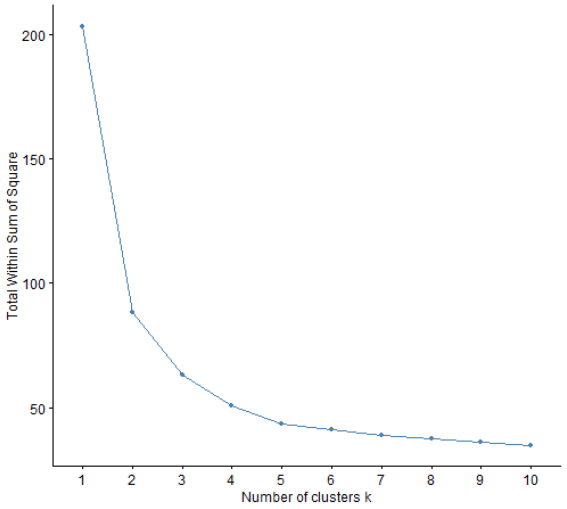


Fig. 272: Total within cluster sum of squares obtained using the c-means algorithm with the Euclidean distance metric when applied to the dataset for the upward ramps.

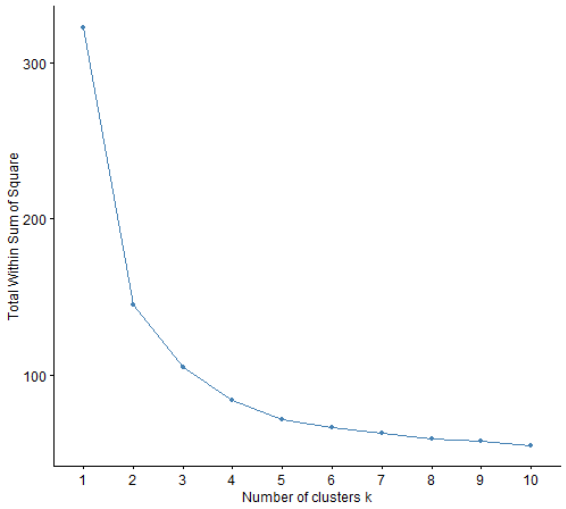


Fig. 273: Total within cluster sum of squares obtained using the c-means algorithm with the Manhattan distance metric when applied to the dataset for the upward ramps.

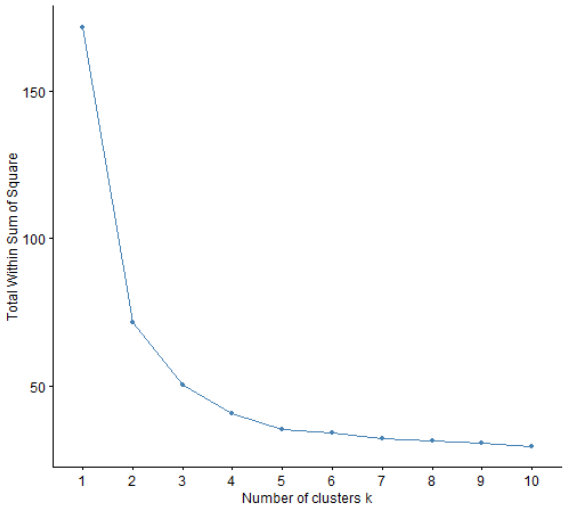


Fig. 274: Total within cluster sum of squares obtained using the c-means algorithm with the maximum distance metric when applied to the dataset for the upward ramps.

Fig. 275 to Fig. 277 plots the average silhouette width as a function of the number of clusters, as obtained with the c-means algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the upward ramps. The silhouette method suggests that the optimal number of clusters to group the upward ramps with the c-means algorithm is $k = 2$ for the Euclidean, Manhattan and maximum distance measures respectively, as shown in Fig. 275 to Fig. 277.

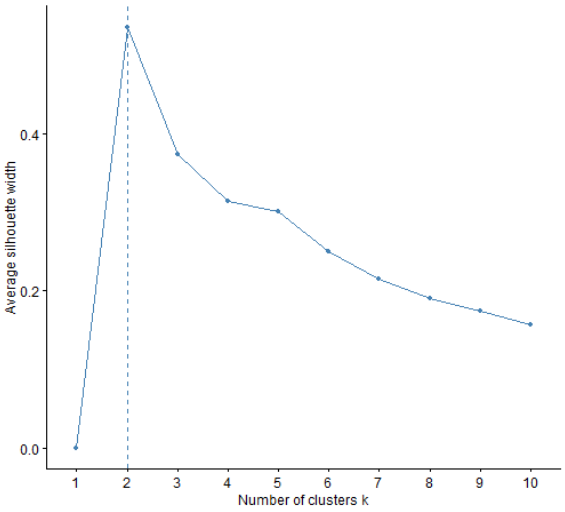


Fig. 275: Average silhouette width obtained using the c-means algorithm with the Euclidean distance metric when applied to the dataset for the upward ramps.

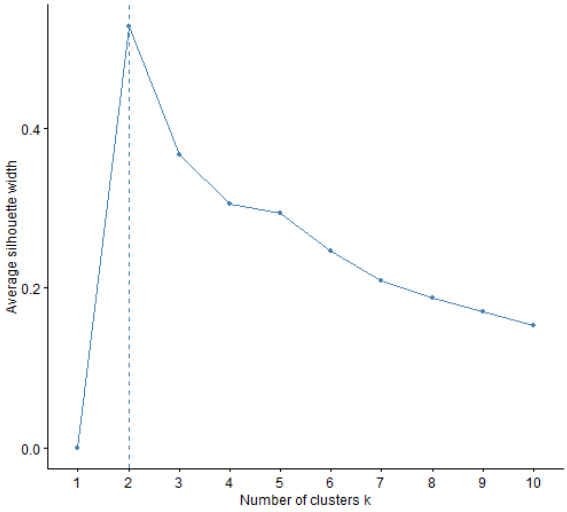


Fig. 276: Average silhouette width obtained using the c-means algorithm with the Manhattan distance metric when applied to the dataset for the upward ramps.

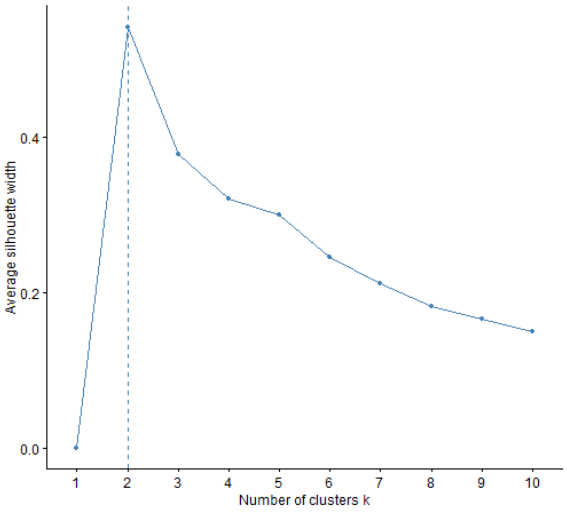


Fig. 277: Average silhouette width obtained using the c-means algorithm with the maximum distance metric when applied to the dataset for the upward ramps.

B.5.2 Downward ramps

Fig. 278 to Fig. 280 plots the total within cluster sum of squares as a function of the number of clusters, as obtained with the c-means algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the downward ramps. From Fig. 278 to Fig. 280, the elbow method suggests that the optimal number of clusters to group the downward ramps with the c-means algorithm is $k = 3$ for the Euclidean, Manhattan and maximum distance measures respectively, since a bend in the curve occurs at $k = 3$. Although further reduction in the total within cluster sum of squares is observed, the change from $k = 3$ to $k = 4$ is insignificant compared to the change from $k = 2$ to $k = 3$.

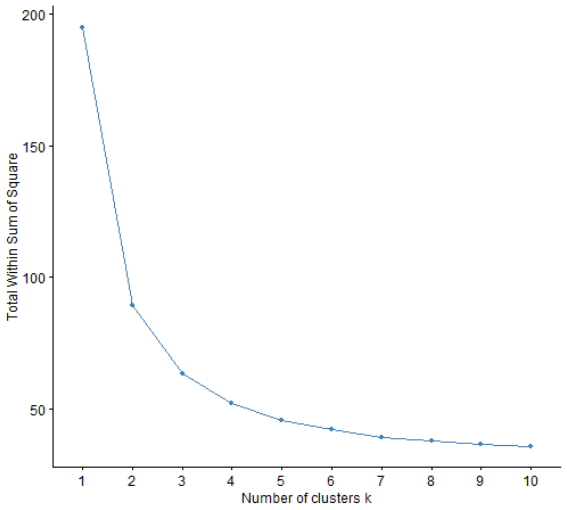


Fig. 278: Total within cluster sum of squares obtained using the c-means algorithm with the Euclidean distance metric when applied to the dataset for the downward ramps.

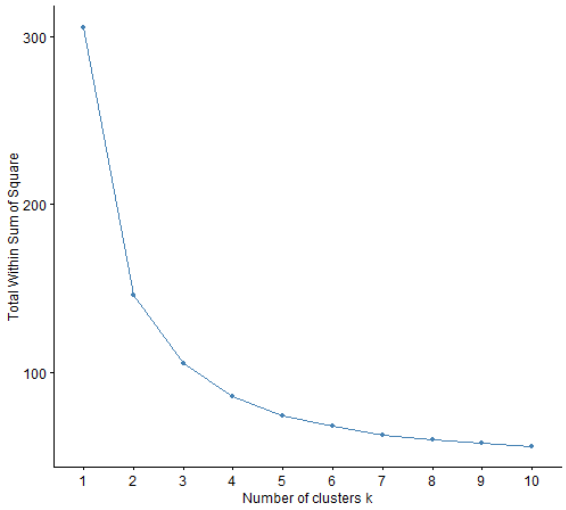


Fig. 279: Total within cluster sum of squares obtained using the c-means algorithm with the Manhattan distance metric when applied to the dataset for the downward ramps.

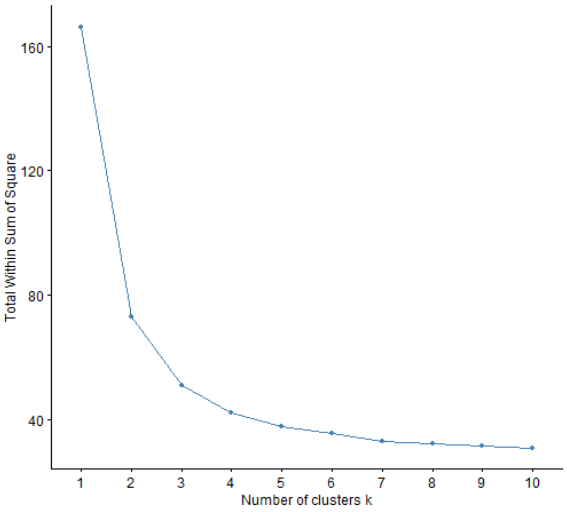


Fig. 280: Total within cluster sum of squares obtained using the c-means algorithm with the maximum distance metric when applied to the dataset for the downward ramps.

Fig. 281 to Fig. 283 plots the average silhouette width as a function of the number of clusters, as obtained with the c-means algorithm for the Euclidean, Manhattan and maximum distance measures respectively, when applied to the dataset for the downward ramps. The silhouette method suggests that the optimal number of clusters to group the downward ramps with the c-means algorithm is $k = 2$ for the Euclidean, Manhattan and maximum distance measures respectively, as shown in Fig. 281 to Fig. 283.

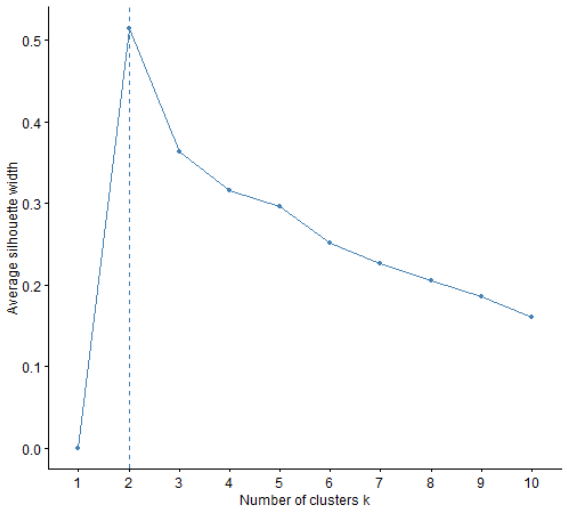


Fig. 281: Average silhouette width obtained using the c-means algorithm with the Euclidean distance metric when applied to the dataset for the downward ramps.

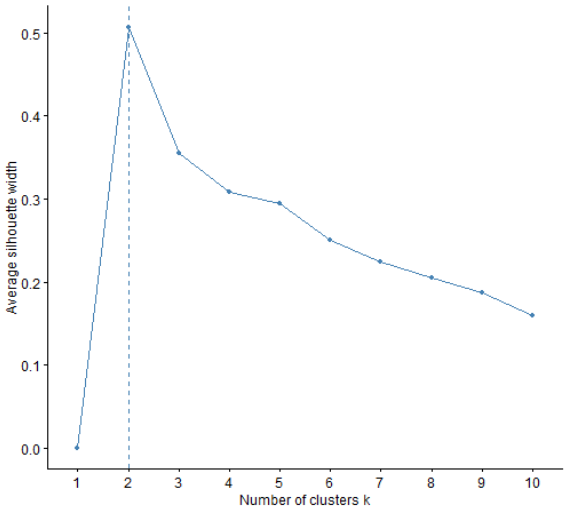


Fig. 282: Average silhouette width obtained using the c-means algorithm with the Manhattan distance metric when applied to the dataset for the downward ramps.

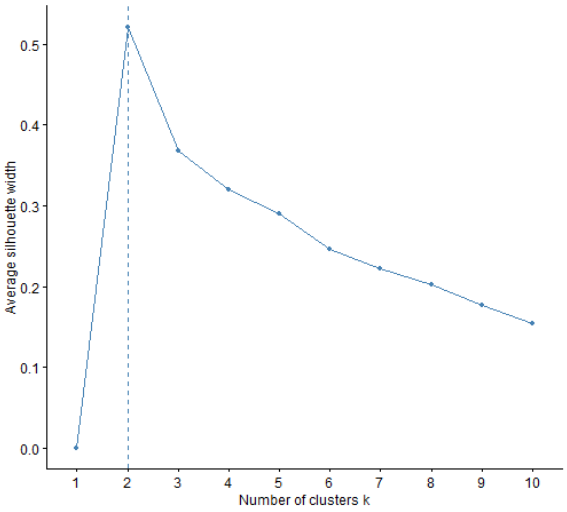


Fig. 283: Average silhouette width obtained using the c-means algorithm with the maximum distance metric when applied to the dataset for the downward ramps.

Appendix C Dendrograms for agglomerative clustering

C.1 Upward ramps

Fig. 284 to Fig. 287 show a dendrogram of the results of the agglomerative algorithm with the Euclidean distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the upward ramps. A dendrogram is a tree-like structure that illustrates the order in which the clusters were merged during each step.

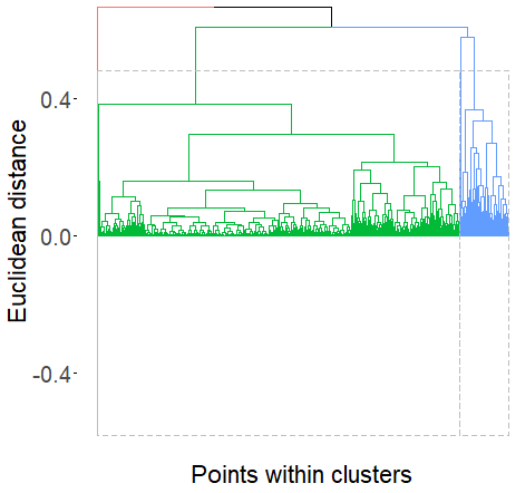


Fig. 284: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the Euclidean distance metric and the average linkage criterion.

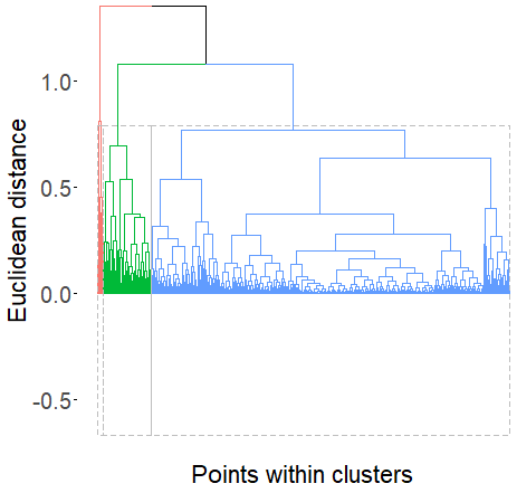


Fig. 285: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the Euclidean distance metric and the complete linkage criterion.

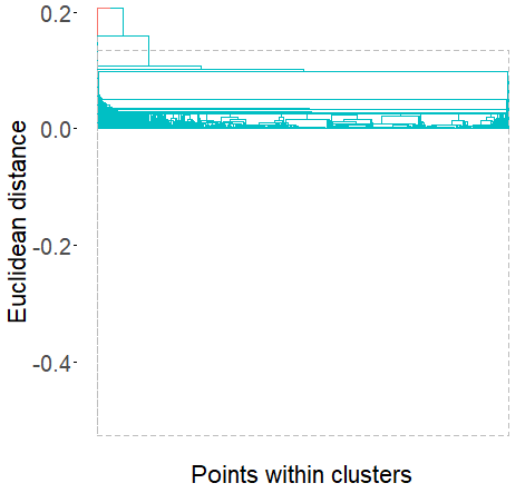


Fig. 286: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the Euclidean distance metric and the single linkage criterion.

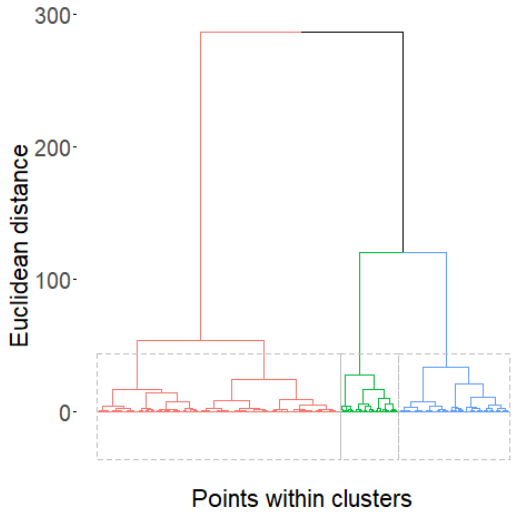


Fig. 287: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the Euclidean distance metric and the Ward linkage criterion.

Fig. 288 to Fig. 291 show a dendrogram of the results of the agglomerative algorithm with the Manhattan distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the upward ramps.

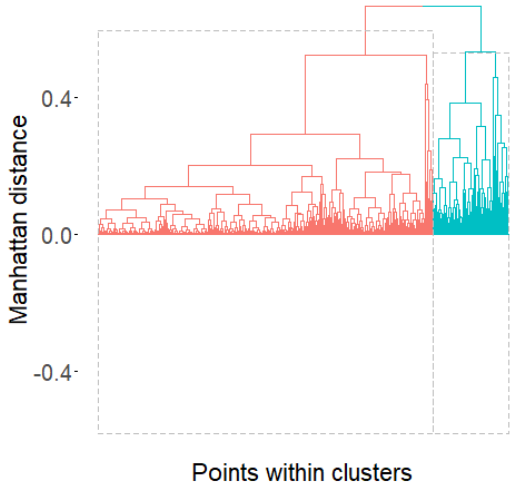


Fig. 288: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the Manhattan distance metric and the average linkage criterion.

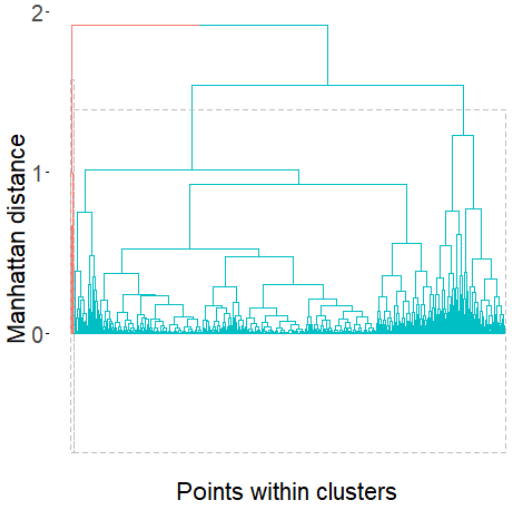


Fig. 289: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the Manhattan distance metric and the complete linkage criterion.

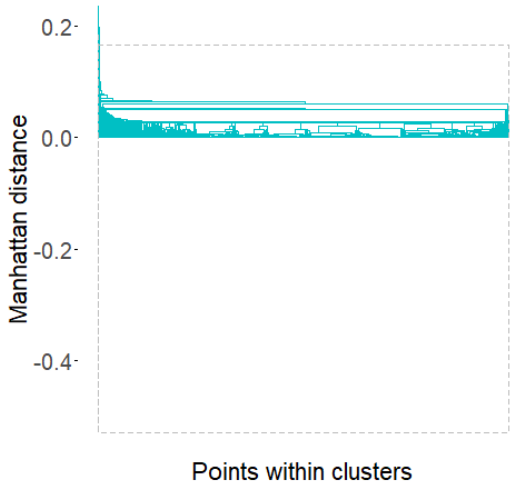


Fig. 290: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the Manhattan distance metric and the single linkage criterion.

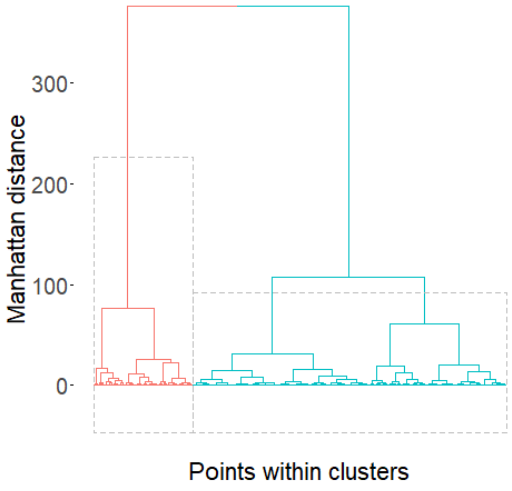


Fig. 291: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the Manhattan distance metric and the Ward linkage criterion.

Fig. 292 to Fig. 295 show a dendrogram of the results of the agglomerative algorithm with the maximum distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the upward ramps.

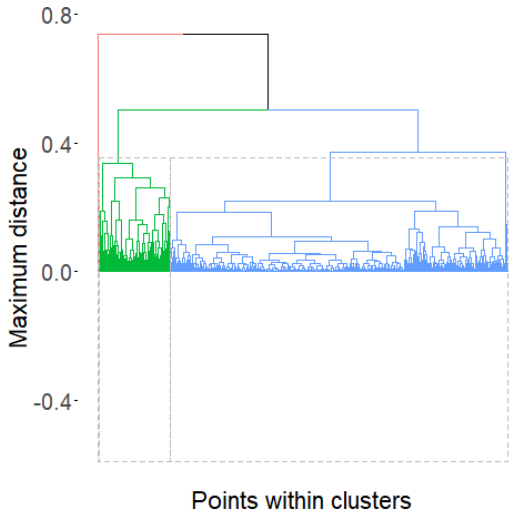


Fig. 292: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the maximum distance metric and the average linkage criterion.

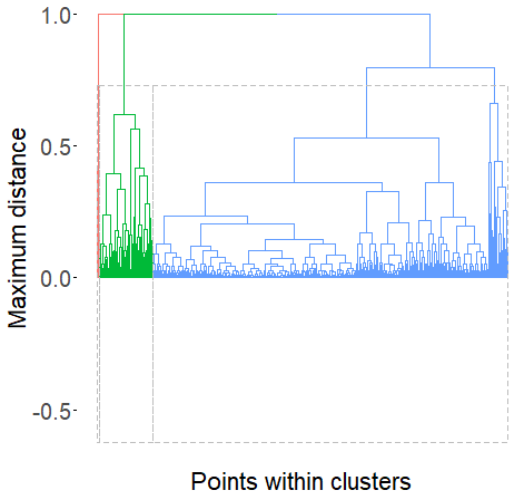


Fig. 293: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the maximum distance metric and the complete linkage criterion.

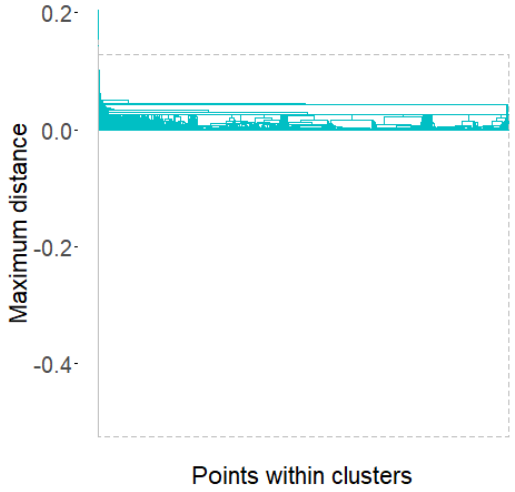


Fig. 294: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the maximum distance metric and the single linkage criterion.

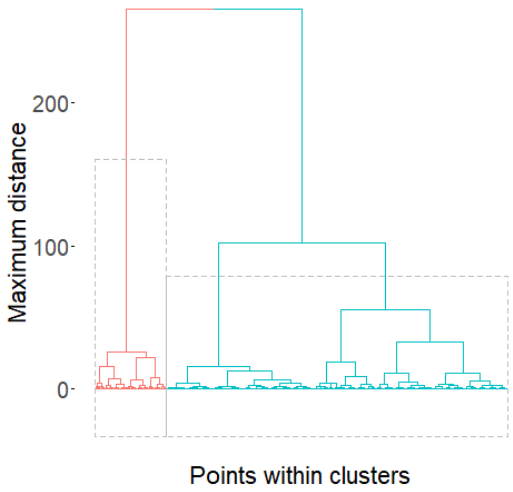


Fig. 295: A dendrogram of the cluster assignments obtained for the upward ramps using the agglomerative algorithm with the maximum distance metric and the Ward linkage criterion.

C.2 Downward ramps

Fig. 296 to Fig. 299 show a dendrogram of the results of the agglomerative algorithm with the Euclidean distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the downward ramps.

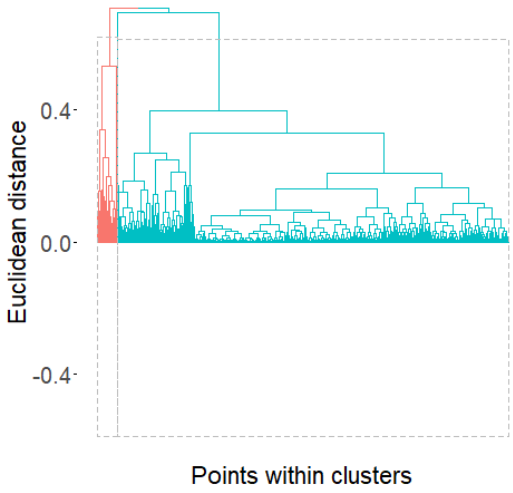


Fig. 296: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the Euclidean distance metric and the average linkage criterion.

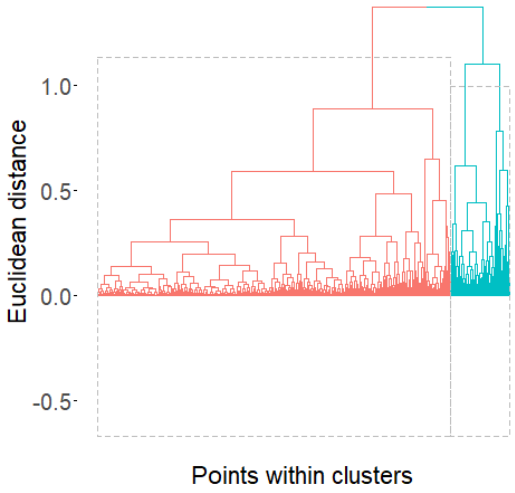


Fig. 297: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the Euclidean distance metric and the complete linkage criterion.

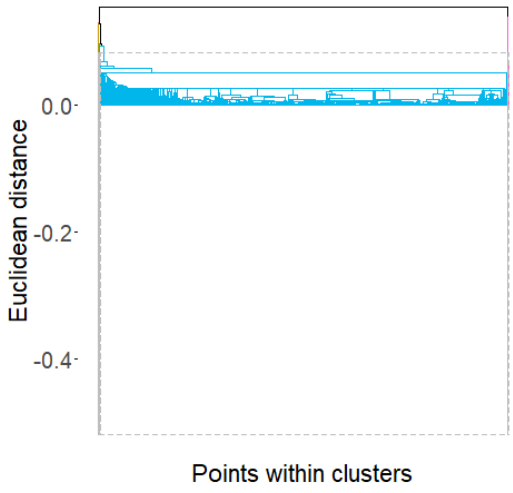


Fig. 298: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the Euclidean distance metric and the single linkage criterion.

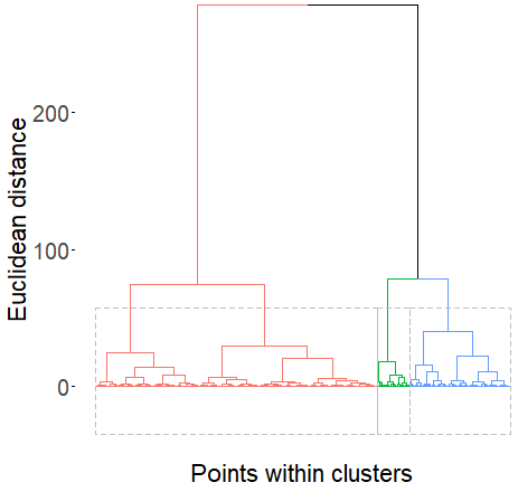


Fig. 299: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the Euclidean distance metric and the Ward linkage criterion.

Fig. 300 to Fig. 303 show a dendrogram of the results of the agglomerative algorithm with the Manhattan distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the downward ramps.

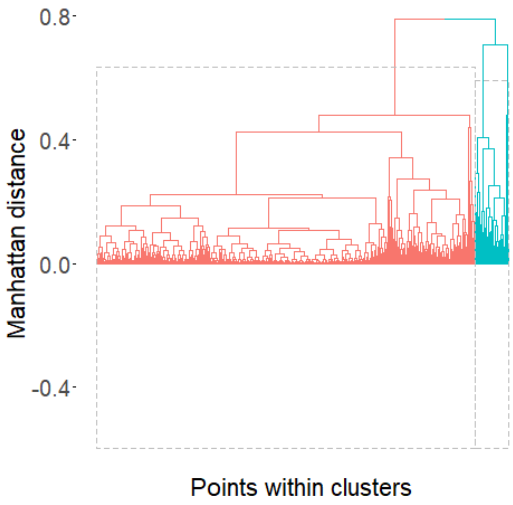


Fig. 300: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the Manhattan distance metric and the average linkage criterion.

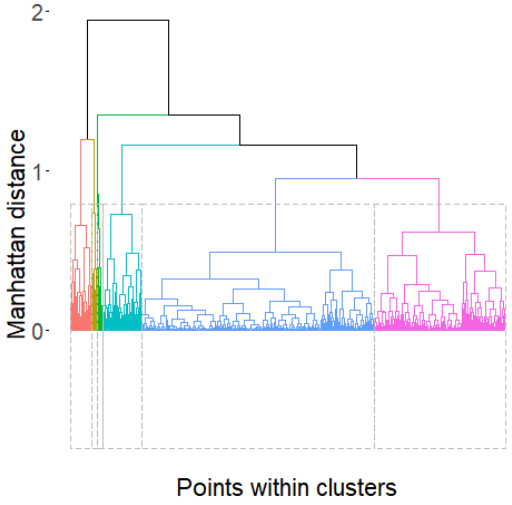


Fig. 301: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the Manhattan distance metric and the complete linkage criterion.

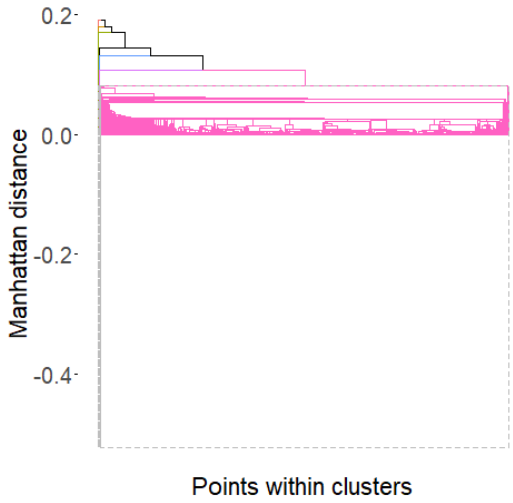


Fig. 302: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the Manhattan distance metric and the single linkage criterion.

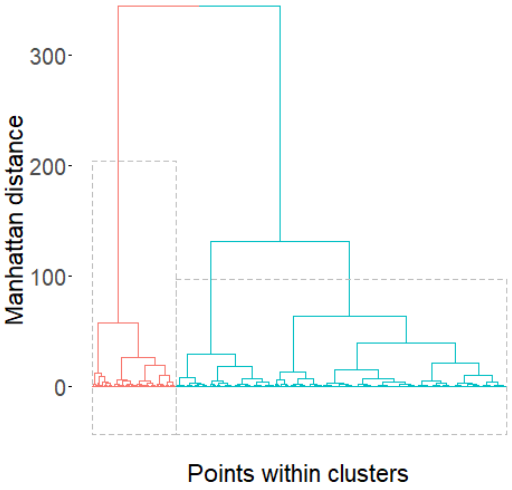


Fig. 303: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the Manhattan distance metric and the Ward linkage criterion.

Fig. 304 to Fig. 307 show a dendrogram of the results of the agglomerative algorithm with the maximum distance metric and the average, complete, single and Ward linkage criteria, respectively, as applied to the dataset for the downward ramps.

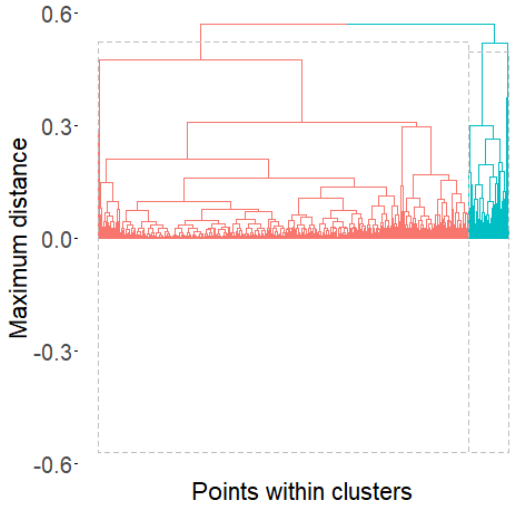


Fig. 304: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the maximum distance metric and the average linkage criterion.

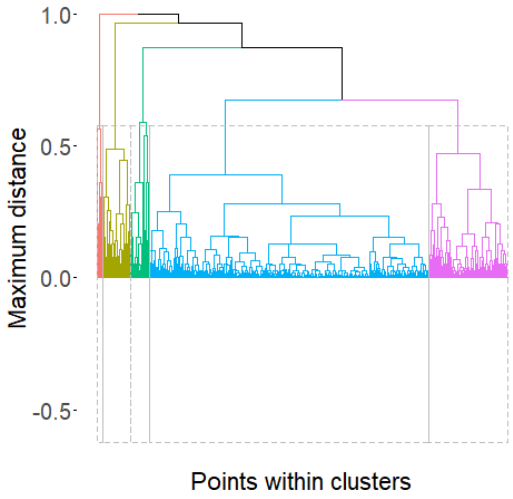


Fig. 305: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the maximum distance metric and the complete linkage criterion.

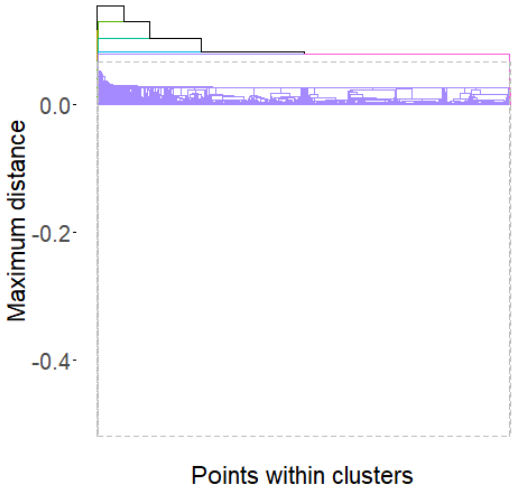


Fig. 306: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the maximum distance metric and the single linkage criterion.

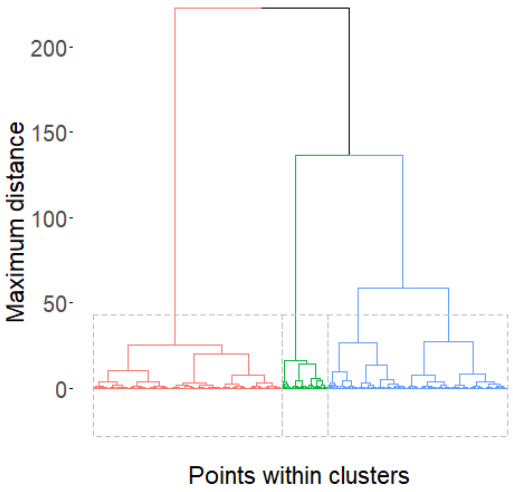


Fig. 307: A dendrogram of the cluster assignments obtained for the downward ramps using the agglomerative algorithm with the maximum distance metric and the Ward linkage criterion.